150 ptas.
mi COMPUTER
14
CURSO PRACTICO DEL ORDENADOR PERSONAL,
EL MICRO Y EL MINIORDENADOR
File/Print Edit Search Type Style Format Page Layout
261 El Lisa de Apple
264 Videotex
266 Juegos postales
267 Simulación
269 Generadores de caracteres
270 Tandy Color
272 Programación Basic
276 Sonido y luz
278 Visualizaciones en cristal líquido
Lisa
Editorial Delta, S.A.

MI COMPUTER CURSO PRACTICO
DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Publicado por Editorial Delta, S.A., Barcelona, y comercializado en exclusiva por Distribuidora Olimpia, S.A., Barcelona

Volumen II - Fascículo 14

Director: José Mas Godayol
Director editorial: Gerardo Romero
Jefe de redacción: Pablo Parra
Coordinación editorial: Jaime Mardones
Asesor técnico: Jesús Nebra

Redactores y colaboradores: G. Jefferson, R. Ford, S. Tarditti, A. Cuevas
Para la edición inglesa: R. Pawson (editor), D. Tebbutt (consultant editor), C. Cooper (executive editor), D. Whelan (art editor), Bunch Partworks Ltd. (proyecto y realización)

Realización gráfica: Luis F. Balaguer

Redacción y administración:
Paseo de Gracia, 88, 5.°, Barcelona-8
Tels. (93) 215 10 32 / (93) 215 10 50 - Télex 97848 EDLTE

MI COMPUTER, *Curso práctico del ordenador personal, el micro y el miniordenador,* se publica en forma de 96 fascículos de aparición semanal, encuadernables en ocho volúmenes. Cada fascículo consta de 20 páginas interiores y sus correspondientes cubiertas. Con el fascículo que completa cada uno de los volúmenes, se ponen a la venta las tapas para su encuadernación.
El editor se reserva el derecho de modificar el precio de venta del fascículo en el transcurso de la obra, si las circunstancias del mercado así lo exigieran.

ISBN: 84-85822-83-8 (fascículo) 84-85822-90-0 (tomo 2)
84-85822-82-X (obra completa)
Depósito Legal: B. 52/1984

Fotocomposición: Tecfa, S.A., Pedro IV, 160, Barcelona-5
Impresión: Cayfosa, Santa Perpètua de Mogoda (Barcelona) 188404
Impreso en España - Printed in Spain - Abril 1984

Editorial Delta, S.A., garantiza la publicación de todos los fascículos que componen esta obra.

Distribuye para España: Marco Ibérica, Distribución de Ediciones, S.A., Carretera de Irún, km 13,350. Variante de Fuencarral, Madrid-34.
Distribuye para Argentina: Viscontea Distribuidora, S.C.A., La Rioja 1134/56, Buenos Aires.
Distribuye para Colombia: Distribuidoras Unidas, Ltda., Transversal 93, n.° 52-03, Bogotá D.E.
Distribuye para México: Distribuidora Intermex, S.A., Lucio Blanco, n.° 435, Col. San Juan Tlihuaca, Azcapotzalco, 02400, México D.F.
Distribuye para Venezuela: Distribuidora Continental, S.A., Ferrenquín a Cruz de Candelaria, 178, Caracas, y todas sus sucursales en el interior del país.

Pida a su proveedor habitual que le reserve un ejemplar de MI COMPUTER. Comprando su fascículo todas las semanas y en el mismo quiosco o librería, Vd. conseguirá un servicio más rápido, pues nos permite realizar la distribución a los puntos de venta con la mayor precisión.

Servicio de suscripciones y atrasados (sólo para España)

Las condiciones de suscripción a la obra completa (96 fascículos más las tapas, guardas y transferibles para la confección de los 8 volúmenes) son las siguientes:

a) Un pago único anticipado de 16 690 ptas. o bien 8 pagos trimestrales anticipados y consecutivos de 2 087 ptas. (sin gastos de envío).

b) Los pagos pueden hacerse efectivos mediante ingreso en la cuenta 3371872 de la Caja Postal de Ahorros y remitiendo a continuación el resguardo o su fotocopia a Distribuidora Olimpia (Paseo de Gracia, 88, 5.°, Barcelona-8), o también con talón bancario remitido a la misma dirección.

c) Se realizará un envío cada 12 semanas, compuesto de 12 fascículos y las tapas para encuadernarlos.

Los fascículos atrasados pueden adquirirse en el quiosco o librería habitual. También pueden recibirse por correo, con incremento del coste de envío, haciendo llegar su importe a Distribuidora Olimpia, en la forma establecida en el apartado b).

Para cualquier aclaración, telefonear al (93) 215 75 21.

No se efectúan envíos contra reembolso.



Ordenador del futuro

El Lisa de Apple es el ordenador de gestión más innovador que existe en el mercado, y muchas de sus configuraciones aparecerán finalmente en los ordenadores personales

El Lisa, fabricado por Apple Computer, es un ordenador destinado exclusivamente a ser utilizado con fines de gestión empresarial. Vale dos millones de pesetas, sin incluir la impresora. El lector tal vez se pregunte por qué se le otorga tanta importancia en esta obra a una máquina tan cara. La razón por la que le hemos concedido tanta atención se explica por el hecho de que el Lisa está tan adelantado a su época, que es seguro que muchas de sus configuraciones finalmente se aplicarán a los futuros ordenadores personales. Se sabe que la Apple ya está trabajando con versiones a escala reducida del proyecto, y los fabricantes de la competencia se apresuran en imitar las características de este innovador ordenador.

No es el hardware del Lisa lo que es tan radicalmente nuevo, sino más bien el software estándar que viene con él. Desarrollar software sofisticado es, de hecho, una tendencia en la que en los momentos actuales coinciden todos los microordenadores. Ahora, diseñar y construir un nuevo tipo de microordenador exige menos horas de trabajo que escribir un programa sofisticado de gestión o de juego recreativo. El software se está convirtiendo en el elemento más importante de cualquier sistema informático, así como en el más caro. En la actualidad los usuarios de ordenadores personales se encuentran con que, en el transcurso de un año, han invertido en la adquisición de programas en cassette y en cartuchos la misma cantidad de dinero que destinaron a la compra de la máquina.

No obstante, primero analizaremos el hardware del Lisa, cuyo diseño fue dictado fundamentalmente por los requerimientos del software. El Lisa viene con un megabyte de RAM como memoria estándar (es decir, mil veces la de un ZX81). Tal enorme cantidad de memoria exige que el microprocesador dedique mucho tiempo a la función de "administración de memoria": desplazando los datos por la memoria y llevando el registro de dónde se encuentra cada elemento. El procesador es un Motorola 68000, que es un dispositivo de 16 bits (esto significa que es capaz de procesar 16 bits de datos simultáneamente, mientras que la mayoría de las CPU de los ordenadores personales manipulan 8). Desde el punto de vista de los ordenadores personales, se trata de un procesador muy rápido, con un conjunto de instrucciones muy avanzado. Para el almacenamiento permanente, el sistema Lisa incluye dos unidades de disco flexible y una unidad de disco rígido separada y con pocas características externas. El disco rígido es necesario tanto por su capacidad (cinco megabytes) como por su velocidad: el Lisa hace uso de una gran cantidad de programas que frecuentemente se necesita intercambiar entre la RAM y el disco.

Otra característica notable del hardware del Lisa es la visualización monocromática incorporada, que posee una resolución de 720 × 364 pixels. Ello permite una variedad de caracteres distintos para el texto, así como el tipo de gráficos que vemos en este artículo. El diseño del Lisa incorpora chips y sistemas de circuitos especiales destinados exclusivamente a accionar esta visualización y a mover rápidamente las imágenes.

En conexión con una impresora adecuada (como una unidad matricial de gran velocidad y gran calidad), se puede reproducir en papel todo cuanto se visualiza en pantalla. Sin embargo, aunque la impresora no fuera compatible con el alto nivel de resolución de la pantalla, el Lisa produciría una imagen impresa de la mejor calidad.

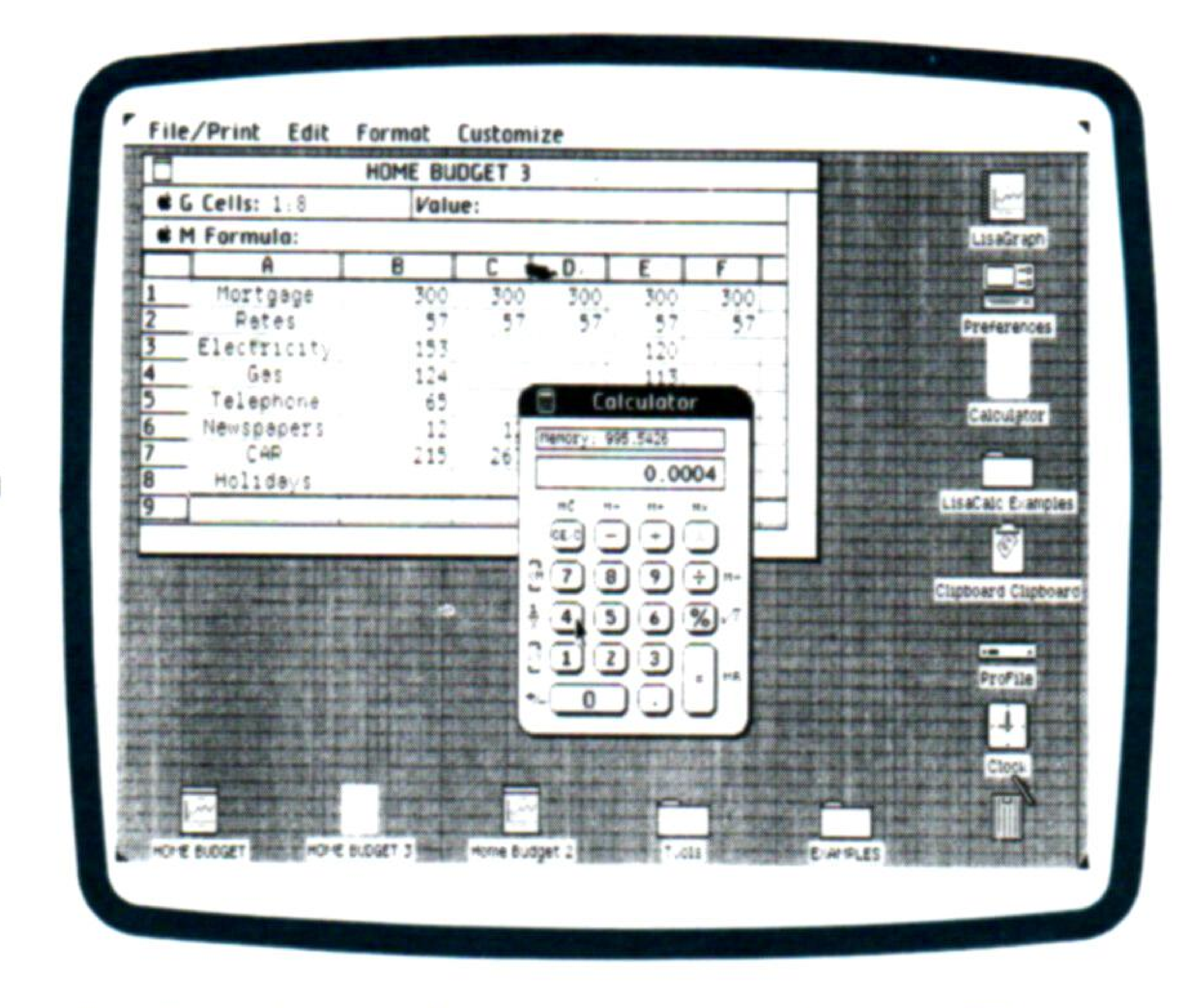
El teclado del Lisa se puede separar de la unidad principal y tiene un buen trazado. No obstante, se utiliza con muchísima menos frecuencia que el de otras máquinas, porque el Lisa posee un "ratón". Un "ratón" representa una de las diversas maneras de dar entrada a información en la pantalla sin necesidad de valerse del teclado; otros métodos son la palanca de mando, el lápiz óptico y las unidades para reconocimiento de voz. En esencia, el "ratón" es una pequeña caja que se mueve a través de la superficie de su mesa o escritorio y que se conecta



Ian McKinnell

Facilidades para el usuario
El Lisa de Apple se diseñó para que lo emplearan personas del campo empresarial que no tuvieran ninguna experiencia previa con ordenadores. La utilización de un "ratón" manual determina que se prescinda del teclado con muchísima mayor frecuencia que en el caso de otros sistemas

La imagen perfecta
Toda función que realice con el Lisa se representa mediante un símbolo denominado "icono". Para activar la función, se mueve el "ratón" hasta que el cursor quede situado sobre el icono, y se pulsa el botón SELECT del "ratón". Entonces "se descubre" la aplicación, visualizándose en la pantalla con todo detalle



al ordenador mediante un cable colgante. El movimiento del "ratón" origina el desplazamiento de un "cursor" o indicador alrededor de la pantalla. De este modo se puede señalar en ésta la información o la orden que usted requiera y luego, pulsando el botón SELECT de la parte superior del ratón, seleccionar esa información o ejecutar esa orden. Sólo es necesario usar el teclado cuando se han de entrar nuevos datos en forma de texto o números.

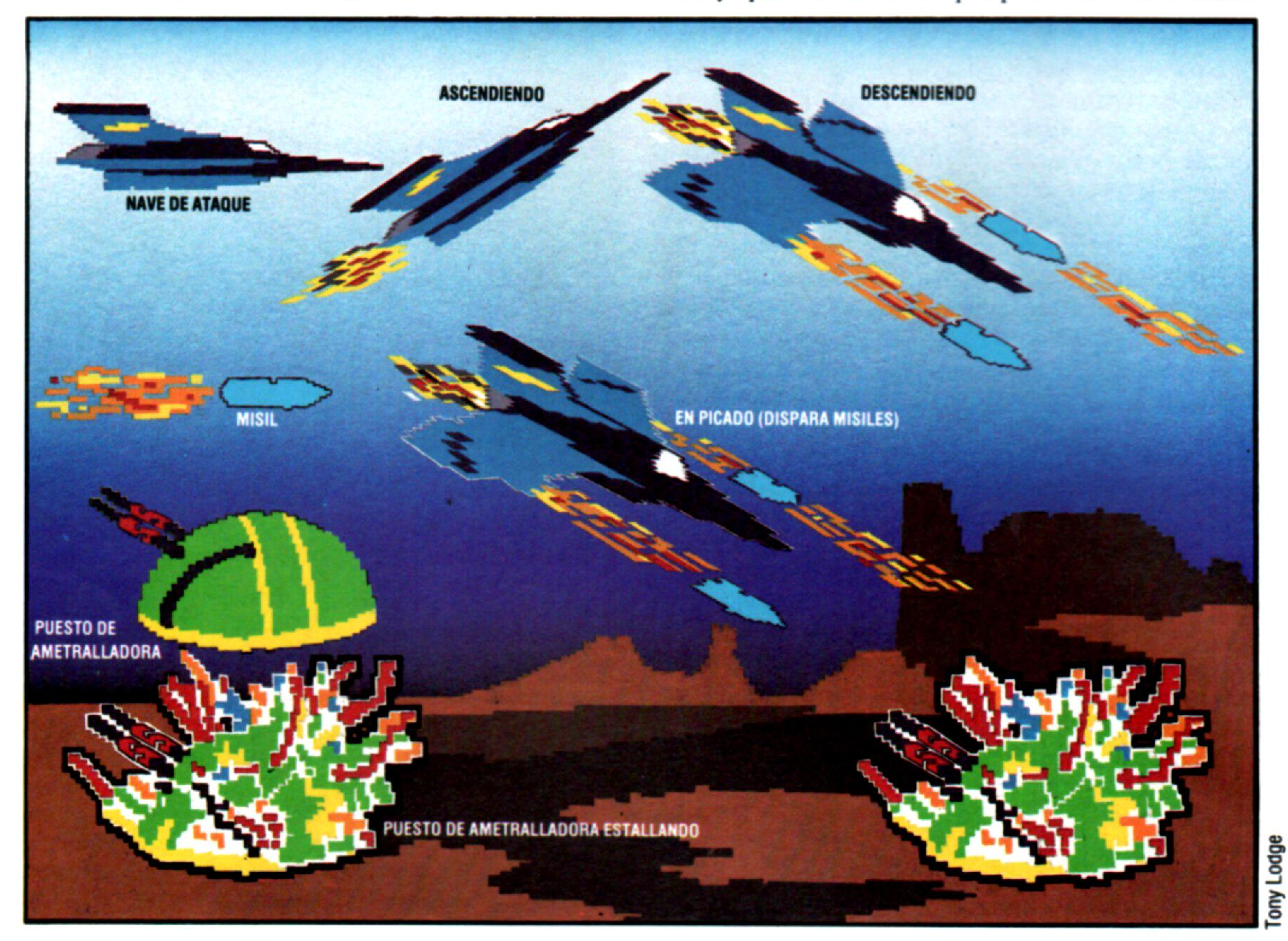
Llegados a este punto, ya estamos preparados para empezar a analizar el software del Lisa. De nuevo deseamos resaltar que, si bien sus aplicaciones se inscriben por completo dentro del campo de la gestión empresarial, los principios sobre los cuales funciona se introducirán con el tiempo en las configuraciones para ordenadores personales.

Cuando se conecta el Lisa por primera vez, la imagen que muestra la pantalla corresponde a la superficie de un escritorio con distintos objetos dispuestos sobre él. De hecho, casi todo lo que usted realice en el Lisa se representa visualmente en la forma en que lo llevaría a cabo si no contara con la ayuda de un ordenador. Ésta es la razón fundamental por la cual a los principiantes les resulta mucho más fácil llegar a familiarizarse con el Lisa que con el hardware y el software convencionales. Los objetos ordenados sobre el escritorio se denominan *iconos*; cada uno representa una función determinada, que por lo general aparece escrita debajo de él.

Tomemos como ejemplo el icono del reloj. Desplazando el cursor sobre la pequeña representación horaria mediante el "ratón" y pulsando el botón SELECT, se visualizará en la pantalla un reloj de tamaño mucho mayor, junto con la fecha del día. Si no desea que el gran reloj le desordene el escritorio, sencillamente lo puede "reducir" otra vez a su tamaño original. Del mismo modo, seleccionar el icono en forma de calculadora hará que aparezca una más grande, que se puede utilizar para resolver operaciones aritméticas sencillas. Si no encuentra satisfactoria la disposición de los iconos sobre el escritorio, con sólo pulsar el botón SELECT y desplazar el "ratón", puede hacerlos cambiar de lugar en la pantalla. Uno de los rasgos más graciosos del Lisa y que ilustra hasta qué punto este ordenador está modelado de acuerdo con nuestros hábitos de trabajo, es el icono que representa el cubo de la basura. Si ya no necesita una parte de la labor realizada, puede simplemente tirarla al cubo de la basura empleando el "ratón". Procediendo así es muy difícil que por accidente se borre alguna información importante. Incluso se puede examinar lo que contiene en ese instante el cubo y recuperarlo, ¡a menos que el ordenador haya sido desconectado!

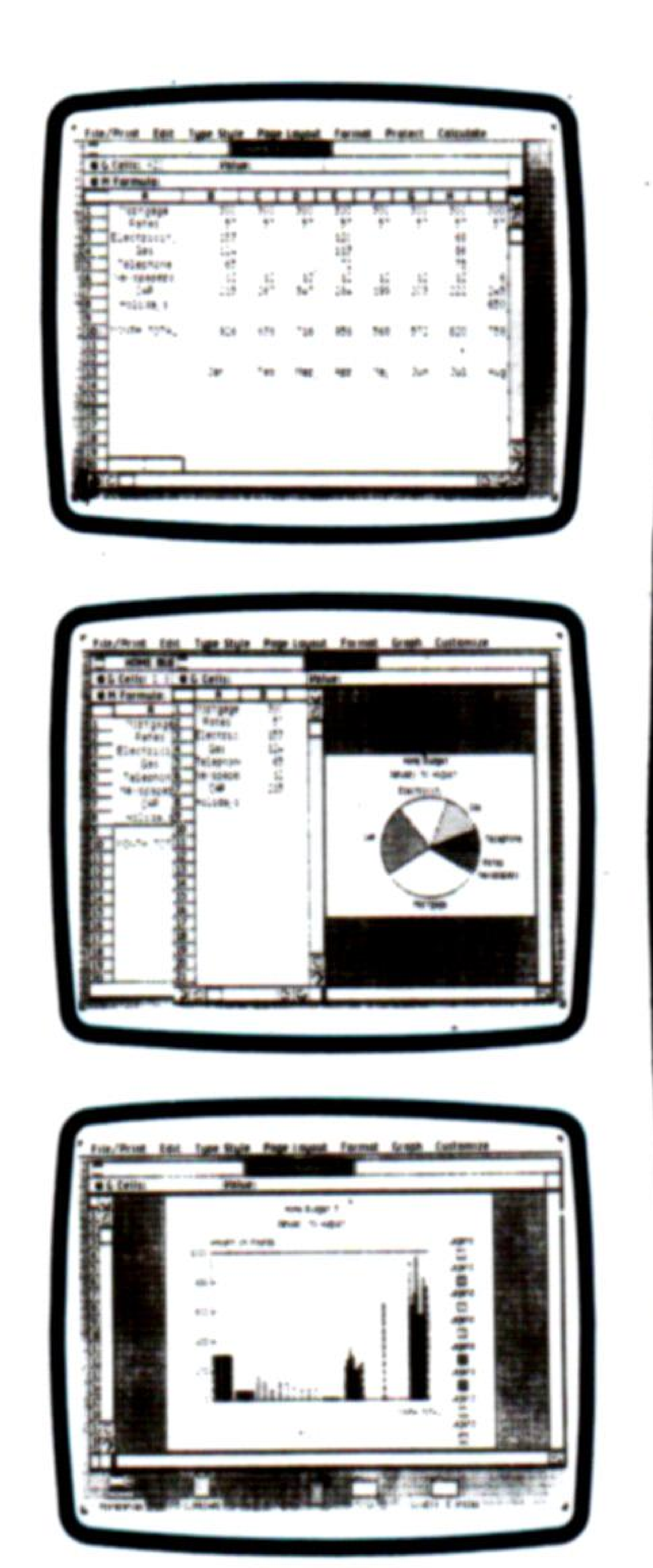
En el Lisa la mayor parte del trabajo real se realiza utilizando uno de los siguientes seis sistemas aplicativos: LisaWrite, un procesador de textos; LisaCalc, una hoja electrónica; LisaGraph, un sis-

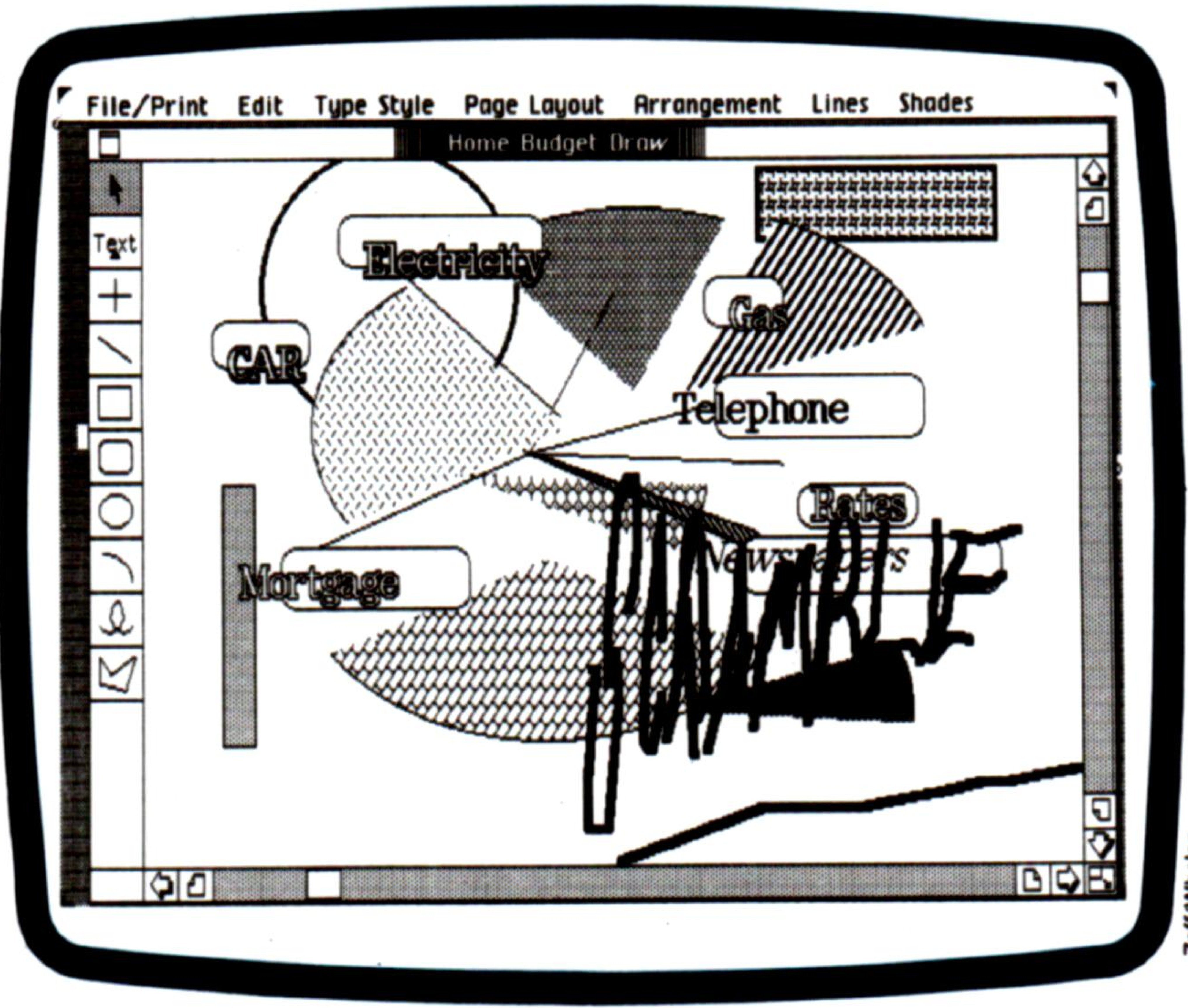
Programación orientada a un objeto
Para crear un juego recreativo al estilo del *Defender* empleando una programación convencional, sería necesario diseñar cierta cantidad de muestras de trazados en pantalla y luego, partiendo de cero, escribir un programa que controlara todo el juego. Utilizando la "programación orientada hacia un objeto" (lo cual por el momento es muy difícil de realizar en ordenadores personales porque no existen lenguajes de programación adecuados), el usuario, en cambio, se concentraría en cada elemento del juego individualmente.
Empezando por la nave de ataque, su definición enunciaría que ésta se desplaza siempre de izquierda a derecha; que cuando la palanca de mando se acciona hacia adelante o hacia atrás la nave se mueve, en consecuencia, hacia arriba o hacia abajo; y que cuando se pulsa el botón FIRE, se lanza un misil. Su segunda definición es la relativa al misil. Define su forma y enuncia que continúa en una dirección hasta que entra en contacto con otro objeto, en cuyo punto desaparece. El puesto de ametralladora estática se define mediante una forma sencilla, que se cambia por una imagen de explosión si un misil entra en contacto con ella. ¡Exprese estas tres definiciones en un lenguaje adecuado, colóquelas juntas en serie y el juego estará escrito más o menos para usted!



Tony Lodge

Pasando la información
Una de las configuraciones distintivas del Lisa permite pasar la información de una aplicación a otra mediante la opción COPY. Ésta almacena temporalmente la información en el icono de la tablilla con chips, disponiéndola para "pegarla" en otra ventana. Por ejemplo, podríamos partir del análisis de algunas cifras utilizando LisaCalc (la aplicación para hoja electrónica). Las cifras resultantes estarían en condiciones luego de ser copiadas en LisaGraph, que produciría un diagrama o gráfico de barras de las cifras automáticamente. Por último, la imagen completa se podría transferir a LisaDraw, que nos permitiría adornarla con etiquetas, flechas, diagramas u otros detalles adicionales. Luego se podría imprimir el resultado final





Zeff/Whelan

tema para trazado de gráficos; LisaList, un administrador de base de datos; LisaProject, un medio auxiliar para planificación de proyectos; y LisaDraw, una sofisticada herramienta para crear toda clase de imágenes gráficas. Los iconos para estas aplicaciones son simplemente blocs de papel. Por ejemplo, para efectuar un cálculo por hoja electrónica se coloca el cursor sobre el bloc de papel dividido en líneas y columnas denominado LisaCalc. En efecto, al pulsar el botón SELECT se "arrancará" una hoja de este papel, que se colocará después en algún lugar del escritorio y se le podrá colocar la etiqueta de "planes de venta", por ejemplo.

En realidad el usuario puede optar por tener simultáneamente sobre su escritorio diversas aplicaciones de hoja electrónica a la vez, volviendo a seleccionar el mismo icono. En un sistema por ordenador normal, habría de pasar por el proceso de cargar el programa de hoja electrónica y especificar luego el archivo de datos sobre el que desea trabajar. En el Lisa, sin embargo, el programa y los datos son inseparables. Éste es otro ejemplo de programación orientada hacia un objetivo, que hemos introducido al analizar el Pinball Construction Set (véase p. 241).

Otra característica importante del *entorno operativo* (así se lo denomina) del Lisa es su capacidad para "mirar por la ventana". Cuando se selecciona una aplicación, ésta aparece como una hoja grande de papel dispuesta sobre el escritorio. Las dimensiones de este trozo de papel también se pueden especificar, utilizando el "ratón". Si en un momento dado hay "abierta" a la vez más de una de estas aplicaciones, las hojas se ordenarán una sobre otra, visualizándose en la parte superior la hoja en la que está trabajando en ese momento, tal como ocurriría de estar trabajando en un escritorio. Podría darse el caso de que la aplicación sobre la que esté trabajando, por ejemplo el procesador de textos, requiriera más espacio que el disponible de acuerdo a las dimensiones de la hoja que usted hubiera especificado. En ese caso, la hoja actúa sólo como una ventana a la aplicación y se puede mover alrededor para visualizar cualquier parte del documento total.

Se puede trasladar la información desde una aplicación a otra, empleando otra vez los iconos, y éste es otro de los puntos fuertes del Lisa. Supongamos que usted está haciendo un análisis de sus cifras de ventas mensuales utilizando el LisaCalc. Valiéndose de la función COPY, que se selecciona a partir de un menú de funciones especiales listadas a lo largo de la parte superior de la pantalla, puede hacer una copia temporal de los resultados de la hoja electrónica, que se almacenan en el icono que representa una tablilla con chips. Luego, seleccionando un trozo de papel LisaGraph, se puede dar entrada a esos resultados en la sección INPUT DATA de la aplicación para gráficos, sencillamente seleccionando del menú la opción PASTE. Si se le requiriera, LisaGraph produciría luego un pulcro diagrama (o gráfico de barras y líneas), completamente etiquetado y sombreado. Ahora, utilizando nuevamente las opciones COPY y PASTE de la parte superior de la pantalla, se puede copiar esta imagen en un trozo de papel LisaDraw. Esta última aplicación permitiría entonces adornar el diagrama con algunas flechas, o modificar las etiquetas y los encabezamientos según una variedad de tipos diferentes. El resultado final se podría así imprimir y copiar para emplearlo, por ejemplo, como una ilustración para un reportaje o un artículo para alguna revista.

Como hemos dicho, tanto los principios de la programación orientada hacia un objetivo como los de los entornos operativos al estilo Lisa, pronto comenzarán a introducirse incluso en las máquinas más económicas, especialmente en la medida en que éstas se vuelvan más sofisticadas en términos de velocidad de procesamiento y dimensiones de la memoria RAM. Imagínese que en la pantalla de su ordenador personal tuviera múltiples ventanas: podría escribir un programa en una de ellas y observar en otra su salida. Luego podría llamar a otros medios auxiliares de programación con sólo señalar un icono con forma de caja para herramientas, y mover las subrutinas a través de su listado simplemente desplazando el "ratón".

Ventanas al mundo

El videotex es uno de los pocos campos de la informática en que se han fijado estándares internacionales. Ello permite a cualquier ordenador acceder a una red mundial de bases de datos

Los sistemas de videotex, como el servicio Prestel de la British Telecom, permiten tener acceso a una diversidad de bases de datos por medio de un televisor adaptado y un teléfono. Un sistema puede ser de naturaleza pública, abierto y accesible a todos, como el Prestel; puede estar abierto a todos pero tener un contenido más bien especializado, no comprensible a cualquier usuario (como el Fintel, una base de datos financieros que se actualizan de forma constante), o, finalmente, ser de naturaleza privada.

No obstante, los sistemas de videotex no se deben confundir con los sistemas de teletexto que están ahora disponibles a través de los canales de las emisoras de televisión, que utilizan juegos de caracteres y trazados de páginas similares. Los sistemas de teletexto, como el Ceefax de la BBC o el Oracle de la ITV (cadena privada de televisión), se emiten como señales subsidiarias superpuestas a los programas de televisión normales. Las señales que definen a una página de teletexto se transmiten entre los fotogramas de una emisión corriente de televisión. Un dispositivo decodificador las separa y crea la imagen de teletexto. El texto y los gráficos resultantes pueden desplazar a la imagen de emisión normal o se pueden superponer a ella de manera que aparezcan las dos juntas, como cuando se utiliza el teletexto para proporcionar subtítulos para no interferir el sonido.

Los sistemas de teletexto no son interactivos, es decir, el espectador no tiene forma alguna de modificar los contenidos de una página de información, ni siquiera de dar una respuesta. Sin embargo, el acceso a ellos es gratuito. Lo único que se necesita para acceder a un gran volumen de información útil es un televisor convenientemente adaptado.

Los sistemas de videotex, por el contrario, utilizan el televisor doméstico simplemente como un monitor para visualizar los datos recibidos a través de la red telefónica. Normalmente, los usuarios no pueden modificar la base de datos, pero pueden utilizar su teclado para dar entrada a preguntas en el sistema de respuestas. Todos los sistemas de videotex son activados por menú, es decir, siempre se visualiza la gama de opciones de que dispone el usuario a cualquier nivel. El usuario realiza su elección dando entrada a un número en el teclado numérico. Éste transfiere el control del submenú seleccionado; y el procedimiento se sigue repitiendo a través de los niveles jerárquicos hasta que el usuario llega a la página de información deseada.

Cada página se identifica mediante un número exclusivo y se puede llegar directamente a una página determinada dando entrada a este número. Éste es el método más rápido (y más barato) para aquellas personas que consultan con frecuencia una misma página dedicada a tratar cierto tema.

A modo de ejemplo, sigamos los pasos que daría un usuario hipotético para hacer la reserva de sus vacaciones de verano vía Prestel. El cuadro de *sign-on* (primera imagen que se ve cuando se conecta el sistema) le indica que pulse el símbolo # en el teclado para llegar al índice principal. A partir de entonces el usuario sigue las indicaciones del menú página a página. La primera parada es en el cuadro de Información General. La cuarta entrada de esta página está encabezada por la leyenda Vacaciones, transporte, viaje, de modo que digita 4.

En respuesta se le ofrecerá una opción entre Viaje en tren, Viaje en avión, Viaje en autocar, Otros transportes, Vacaciones y turismo y Coches y vehículos motorizados. También hay cuatro "salidas", cada una de las cuales lleva a un cuadro de información (una



Información pública
Al servicio de videotex Prestel de la British Telecom, que consta de más de 250 000 páginas de información, se puede acceder desde todos los rincones de Gran Bretaña, incluso desde un teléfono público en ciertas condiciones. Todo lo que se necesita para consultar la base de datos es una línea telefónica y un receptor de televisión

Cortesía de Prestel



Estado de cuentas en pantalla
La Nottingham Building Society, en asociación con el Bank of Scotland y la British Telecom, han desarrollado un sistema que les permite a los suscriptores controlar muchas de sus operaciones financieras directamente desde sus propios hogares, utilizando un adaptador Prestel que les proporciona el consorcio. Los clientes pueden examinar el estado de sus cuentas bancarias y de la entidad financiera, pagar facturas, comunicarse con otros suscriptores e incluso comprar una serie de bienes y servicios

Cortesía de Prestel



Cortesía de British Telecom



Agencias de viajes
Una de las aplicaciones comerciales más populares del Prestel se realiza en las agencias de viajes. Las líneas aéreas, en particular, poseen sistemas muy sofisticados de reserva y venta de billetes, si bien ambos funcionan en ordenadores separados. El Prestel les permite a los agentes acceder a la mayoría de estos sistemas y, también, vender billetes y reservar plazas directamente. Además, tal como se aprecia en el grabado de la izquierda, se puede obtener información acerca de las llegadas y salidas en los aeropuertos de Gran Bretaña, así como también de los cambios en los horarios

especie de desvío, como las notas a pie de página de un libro) desde los cuales se puede regresar al cuerpo principal del texto. Después de pulsar la tecla correspondiente a **Vacaciones y turismo**, el futuro viajero ha de decidir si reservará el viaje y el alojamiento por separado o si optará por un programa de vacaciones completo, y esa decisión determinará su ruta de salida de la página. Como método alternativo para seleccionar la página, la Prestel también ha publicado una guía, a partir de la cual se puede efectuar una selección inmediata. Por ejemplo, se le proporciona al usuario una referencia directa de la página que incluye una relación de líneas aéreas o una cadena de hoteles.

Concebido en los laboratorios de investigación de la British Telecom en 1971, el primer sistema de videotex entró en funcionamiento hacia 1976 de forma experimental y quedaría a disposición del público bajo la denominación de Prestel a fines de 1979. British Telecom originalmente había llamado Viewdata (videotex) a su sistema, pero al estipularse legalmente que videotex era un término genérico, se vio obligada a cambiarlo por Prestel.

El videotex representa un caso especial en el sentido de que desde su comienzo se convirtió en un estándar aceptado en todo el mundo. Diversos fabricantes de ordenadores y empresas de telecomunicaciones empezaron a producir sistemas propios que utilizaban los protocolos y las estructuras de información del Prestel. Consecuencia de ello ha sido la creación de una red internacional de bases de datos locales, a las que puede tener acceso cualquier suscriptor.

Los sistemas de videotex exigen una línea telefónica abierta durante todo el tiempo que esté en funcionamiento, y el usuario ha de hacer frente a este gasto además del que representa cualquier conexión con el servicio de videotex. Además, existe la

Red de distribución

El servicio Micronet 800 que funciona en Gran Bretaña, les permite a los suscriptores adquirir software directamente del sistema informático central de Prestel. Para utilizar el sistema es necesario recibir las señales de videotex en un microordenador convencional (y no tan sólo un televisor normal adaptado) y almacenar luego el programa, ya sea en cinta de cassette o en disco, desde donde posteriormente se puede cargar y ejecutar siguiendo los procedimientos normales. El Micronet ofrece alrededor de cien programas libres de cargo, y ofrece muchos más de acuerdo a las tarifas comerciales. Los suscriptores de Micronet también tienen acceso automático al resto de la base de datos de Prestel

posibilidad de una tercera cantidad a pagar por el usuario, determinada por la fuente de datos a que ha tenido acceso, pero este cargo queda supeditado a la voluntad del proveedor de la información. También hay que pagar una pequeña suscripción anual. En un intento por reducir los gastos del usuario, Prestel ha instalado cierto número de "concentradores locales" (líneas troncales telefónicas exclusivas para el sistema) que permiten que un suscriptor que llame desde, por ejemplo, Glasgow a un número de Londres pague la misma tarifa que si se tratase de una llamada local.

El hardware que se requiere para utilizar los servicios de videotex se divide en tres tipos principales. El más sofisticado, y también el más caro, es el terminal de videotex construido especialmente. La mayor parte de los usuarios comerciales lo prefieren. La segunda alternativa consiste en acoplarle un adaptador a un televisor doméstico. Existen diversos adaptadores de este tipo, que van desde un sencillo teclado numérico con dial manual como el de un teléfono, hasta un teclado estilo máquina de escribir, con dial totalmente automático. Pero en el corazón de todos estos dispositivos hay un microordenador exclusivo que decodifica las señales que entran y salen. El tercer método, popular entre los usuarios de microordenadores personales, consiste en comprar software para videotex para un micro estándar. Esta última alternativa viene resultando particularmente atractiva desde la introducción de un servicio conocido como Micronet 800, que les ofrece a los suscriptores la posibilidad de cargar programas para ordenador directamente a través de la línea telefónica. Más aún, estos adaptadores permitirán guardar en disco una página de Prestel, eliminando así la necesidad de una conexión telefónica abierta de manera constante.

Prestel también les permite a los suscriptores enviarse mensajes entre sí gracias a un servicio denominado Mailbox. El suscriptor, al encender su terminal Prestel o, en el caso de que la estuviera usando, al terminar una llamada, recibe la notificación de que hay un mensaje esperando. No es necesario, como se podría pensar, tener acceso a un teclado alfanumérico completo para enviar un mensaje. Por otra parte, se encuentran a disposición del usuario una serie de "modelos de mensaje" que se pueden completar valiéndose sólo de números.

A fines de 1983 había en Gran Bretaña alrededor de 35 000 suscriptores de Prestel, que disponían de más de 250 000 páginas de información; además, un cierto número de empresas y organizaciones utilizaban el videotex para sus propias consultas de base de datos interna.

Juegos postales

No todos los juegos por ordenador requieren tiempos de reacción de fracciones de segundo: en los juegos postales, un movimiento puede durar seis semanas, participando docenas de jugadores

Aunque la mayoría de los nuevos compradores de ordenadores personales declaran que lo hacen con la intención de aprender a programar, no cabe la más mínima duda de que la aplicación más popular para este tipo de máquinas son en realidad los juegos. Tal como hemos comentado en numerosas ocasiones, los juegos por ordenador pueden contribuir tanto a la diversión como a la educación. El usuario no tiene por qué limitarse a los juegos recreativos como los "marcianitos". El ordenador ha planteado una nueva serie de conceptos relativos al juego, como los de aventuras (véase p. 161) y los educativos (véase p. 81), por no hablar de las versiones informatizadas de populares juegos de tablero.

Sin embargo, existe un tipo de juegos por ordenador que no hemos mencionado hasta ahora y del que es posible que usted ni siquiera haya oído hablar. Estableciendo un marcado contraste con los juegos recreativos, que exigen una sincronización y unos reflejos de fracciones de segundo, estos juegos se realizan a una velocidad claramente pedestre, ¡teniendo que esperar semanas para poder efectuar el movimiento siguiente! Y, a diferencia de la mayoría de los otros juegos por ordenador, que son sólo para un jugador, en éstos pueden participar varias docenas de jugadores a la vez.

Nos estamos refiriendo a los juegos postales, es decir, juegos en los que los participantes están diseminados por todo el país (o, en el caso de los juegos internacionales, a lo largo y a lo ancho del mundo) y en los que cada jugador especifica sus movimientos sobre el papel a intervalos predeterminados. Los movimientos se envían por correo a un coordinador de juego, quien los alimenta a un microordenador (la mayoría de los juegos postales no exigen que el jugador posea su propio ordenador personal). El coordinador envía luego una salida impresa del ordenador a cada jugador, que ilustra su propia posición y otras informaciones relacionadas con ella, como puede ser la posición de otros jugadores con los cuales haya entrado en contacto.

Estos juegos por lo general continúan durante muchos meses, cuando no indefinidamente, si bien los jugadores pueden incorporarse a ellos o abandonarlos en el momento que lo deseen. Se suele cobrar una tarifa de incorporación para cubrir los materiales iniciales y el reglamento del juego; después se cobra una tarifa de juego por cada movimiento (en Gran Bretaña por lo común es de 1 libra, o sea, unas 225 pesetas). ¡Estos juegos no son un pasatiempo barato! Un movimiento por semana se considera un juego rápido, mientras que los juegos internacionales pueden implicar una espera de seis semanas entre uno y otro turno.

Un típico juego postal puede contar con varias docenas de jugadores. Si hay más personas interesadas en suscribirse, el coordinador simplemente comienza un segundo juego paralelo al primero, utilizando los mismos programas pero distintos discos de datos en su ordenador.

Los juegos postales existían mucho tiempo antes del nacimiento del ordenador: se trataba de campeonatos postales de ajedrez. Incluso el popular juego de mesa *Diplomacy* (en el que los jugadores representan a siete naciones europeas e intentan dominar el continente estableciendo alianzas entre sí y rompiéndolas) se puede jugar por correo.

La introducción del ordenador para efectuar todo el trabajo de cálculo y administración ha supuesto el incremento del alcance de este tipo de juegos, junto con el de su ingeniosidad y su sofisticación. Algunos de ellos imaginan inmensas galaxias a través de las cuales los jugadores realizan maniobras con sus flotas espaciales; otros ofrecen territorios míticos y reinos guerreros. Son comunes las alianzas entre diez jugadores o más, lo que implica un gran intercambio de correspondencia o de llamadas telefónicas. Una medida de la calidad de estos juegos viene dada por el hecho de que el espectro de edades de los jugadores es mucho más amplio que en otros tipos de juegos por ordenador.



Cortesía de Starlord

Guerras galácticas
Starlord ha sido el primer juego postal por ordenador desarrollado en Gran Bretaña. Lo coordina Mike Singleton utilizando un ordenador Commodore Pet 3032, una unidad de disco rígido de 7,5 megabytes y una impresora matricial en color Integrex. El objetivo de cada jugador consiste en descubrir la estrella Throne y convertirse en emperador. A cada movimiento el jugador recibe un mapa que muestra la zona inmediata a la posición de sus fuerzas y una lista de quienes controlan las estrellas más cercanas. Se precisa un disco de gran capacidad debido al alto número de programas que dirigen conjuntamente el juego y a la gran cantidad de información que se debe mantener para más de 700 jugadores. En las tiendas especializadas se vende una revista denominada *Flagship*, con amplia información acerca de los juegos postales

Modelos de comportamiento

La simulación es una técnica informática que permite experimentar una situación que, de otra forma, sería peligrosa o muy cara.

Uno de los usos más importantes de los ordenadores se produce en el área de la simulación. Se trata de un método de planificación hacia adelante por el cual "se simula" en el ordenador un modelo de la situación a analizar. Un modelo proporciona una visión simplificada de la situación, reteniendo los aspectos significativos del problema y descartando los detalles irrelevantes.

Tomemos el ejemplo de dos vasos, uno de los cuales contiene vino blanco y el otro vino tinto. Si se le agrega al vaso de vino blanco una cucharada del vino tinto, se mezcla bien y después se vuelve a echar una cucharada de la mezcla en el vaso de vino tinto, ¿cuál de los vasos tendrá mayor impureza?

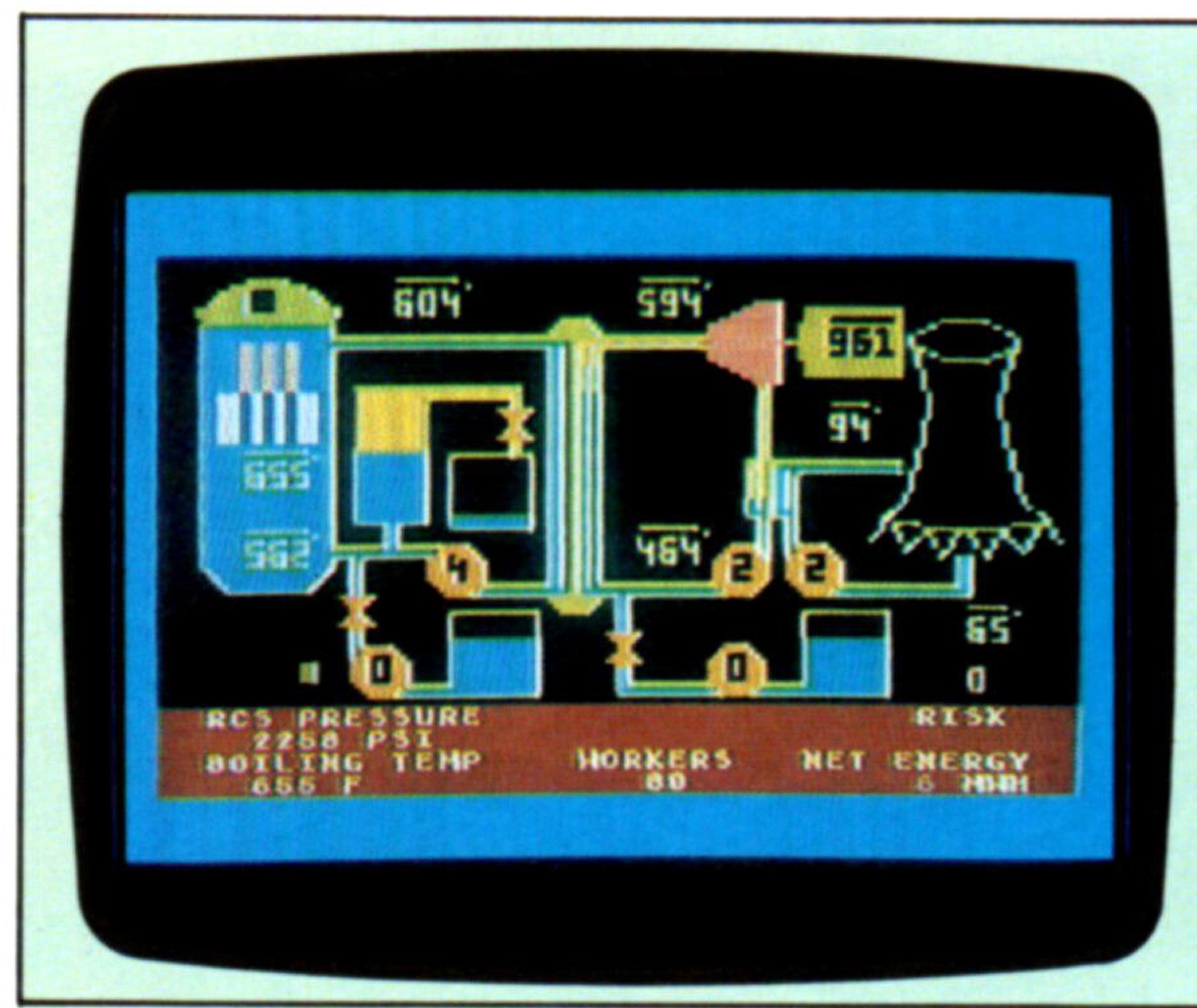
Existen muchas maneras de resolver el problema, pero una de las más sencillas consiste en utilizar un modelo. Por ejemplo, podríamos establecer uno en el que el volumen de vino de cada vaso fuera exactamente de una cucharada. Es fácil comprender que en este caso ambos vasos acabarán con el mismo grado de impureza (igual mezcla de vino tinto y vino blanco). Ampliar nuestro modelo para mayores cantidades de vino nos demostraría que en todos los casos los resultados serían los mismos.

Los modelos aparecen bajo tres formas principales. Un modelo pictórico (por ejemplo, una fotografía o un mapa) muestra acomodaciones y relaciones espaciales entre los elementos que componen esa imagen. Luego están los modelos cuyos componentes se comportan, los unos en relación a los otros, de manera similar a los elementos reales a los que representan en el problema: por ejemplo, los problemas que se resuelven mediante ordenadores analógicos (véase p. 238). El tercer tipo corresponde a los modelos simbólicos que para representar la situación utilizan símbolos abstractos y relaciones matemáticas. Éstos son los que emplean los ordenadores digitales en las simulaciones.

Existen cuatro situaciones principales entre las que podemos optar para resolver un problema mediante la simulación por ordenador. La primera es aquella situación con la cual experimentar podría resultar demasiado peligroso; por ejemplo, determinar qué nivel de radiactividad es inocuo en la zona aledaña a un reactor nuclear.

La segunda situación, de la cual constituye un buen ejemplo un modelo de la economía nacional, corresponde a aquella en la que sería casi imposible hallar una solución puramente matemática para la serie de ecuaciones que componen el problema. Es mejor establecer las ecuaciones en forma de un modelo para ordenador y observar en ellas los efectos de diferentes acciones y acontecimientos.

El tercer caso representa aquella situación en la que el problema a analizar implica tal inversión que todos los ajustes tienen que hacerse efectivos en la etapa de construcción del modelo, antes de que se asuma el compromiso de construir la versión final. En la planificación de una propuesta para un aeropuerto nuevo para la ciudad de Londres, por ejemplo, para los problemas de ingeniería y planificación se llevaron a cabo exhaustivos trabajos de simulación. Esta simulación investigó el ruido y otras consideraciones relativas al medio ambiente, así como el flujo de personas y de tráfico alrededor del emplazamiento propuesto.



Bajo control
Una utilización importante de la simulación se realiza en el campo de la educación, al entrenar a las personas a partir de modelos de sistemas reales. Un ejemplo lo constituye la simulación de una central nuclear, creada por Atari, en la cual el usuario ha de controlar los diversos sistemas de refrigeración para evitar el recalentamiento del reactor. La documentación explica las distintas funciones de control que poseen las centrales nucleares de este tipo, y también dispone de una modalidad de demostración que ejecutará el programa para usted

Software, cortesía de Pilot Software

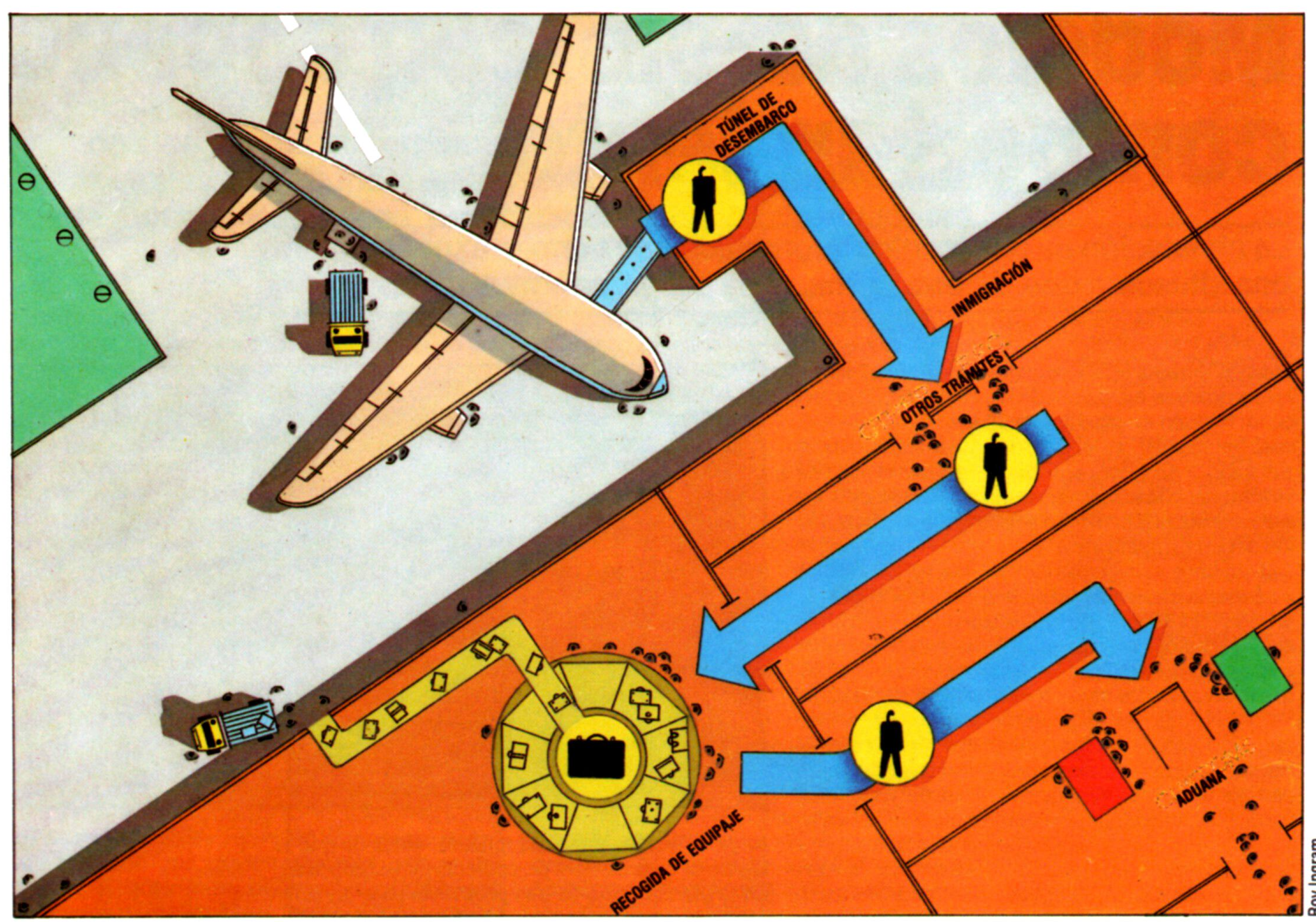
Ian McKinnell

La otra situación en la que resulta de gran valor un modelo es aquella que se traduce en un problema totalmente teórico en el cual es imposible efectuar experimentos físicos. Así, los astrofísicos especulan acerca de cómo nacen las estrellas, y se emplean modelos para evaluar una teoría cosmológica (como la del *big-bang*) en relación a otra.

En toda simulación la primera tarea consiste en construir el modelo. Éste se realiza estudiando la situación y decidiendo cuáles son los elementos importantes y cómo se interrelacionan entre sí.

Los sistemas y sus modelos se dividen en dos familias: los sistemas cerrados o "deterministas" y los sistemas abiertos o "estocásticos". Cuando una familia confecciona el presupuesto para sus gastos, el dinero se puede repartir de muchas maneras, pero al final el balance debe cuadrar, lo que confiere al sistema un carácter determinista. No obstante, si la familia basara cada una de sus decisiones relativas a gastar una suma de dinero en el cara o cruz de una moneda lanzada al aire, y no en función de la cantidad de dinero disponible en su cuenta bancaria, se trataría de un sistema de carácter estocástico.

Con las simulaciones teóricas no se puede estar absolutamente seguro de que el modelo que uno escoja sea el correcto. La gente creía que la Tierra era el centro del universo hasta que Copérnico pro-



Diseño de una terminal aérea
La simulación por ordenador se empleó extensamente en el diseño de la nueva terminal 4 del aeropuerto de Heathrow, en Londres. Primero, el programa hubo de simular los patrones de las llegadas de aviones durante el día, con un número variable de pasajeros y equipaje. Luego simuló los "procesos" por los que habrían de pasar los viajeros. El modelo verificó que el sistema pudiera absorber el volumen de tráfico esperado y sugirió las medidas óptimas de planificación

porcionó un modelo matemático mucho más sencillo, con el Sol en el centro. En la actualidad, al observar lejanas galaxias de estrellas, los astrónomos descubren que todas ellas se alejan de nosotros a velocidades que aumentan de continuo, lo que vuelve a sugerir que nosotros nos hallamos en el centro del universo. Pero si adoptáramos un modelo del universo centrado alrededor de la Tierra estaríamos engañados, tal como demuestra un modelo alternativo. Si se salpica un globo con manchas de tinta que representen las estrellas y después se infla, cada mancha se alejará de las otras y, sin embargo, ninguna de ellas está en el centro del globo.

No obstante, emplear modelos ofrece muchas ventajas: ayudan a formular mejores teorías, aceleran el análisis, permiten intentar modificaciones y, lo más importante de todo, son mucho más baratos que las cosas reales. Su utilización permite, asimismo, realizar saltos creativos en la teoría. El láser, por ejemplo, lo inventó alguien mientras continuaba trabajando en un aspecto de un modelo matemático que previamente se había pasado por alto.

BL Systems, subsidiaria de la British Leyland, comercializa en Gran Bretaña un sistema de simulación para aplicaciones múltiples que originalmente se desarrolló para el diseño de líneas de producción y depósitos automatizados en Cowley y Longbridge. El sistema se denomina *See why* (Vea por qué), y para mostrar los resultados utiliza visualizaciones gráficas en lugar de las listas de estadísticas tradicionales.

Un problema típico que puede afrontar un sistema de simulación es el de hacer cola. En un aeropuerto, por ejemplo, si repentinamente cambia el viento y sólo está disponible una de las pistas de aterrizaje, los aeroplanos han de hacer cola. Los aviones sólo llevan a bordo una reserva de combustible limitada, y para que cada avión aterrice se requiere un tiempo específico. Las reglas para hacer cola se programarán en el sistema. En estas simulaciones se emplean generadores de números aleatorios (véase p. 209) para crear acontecimientos inesperados, como llegadas de aviones al azar.

La simulación es un área aplicativa muy importante para los ordenadores digitales e incluso ha dado lugar a la creación de nuevos lenguajes que se han escrito especialmente para proyectos de simulación (por ejemplo, GASP, SIMSCRIPS y GPSS).

Cortesía de BL Systems

"See why" (Vea por qué)
BL Systems, subsidiaria de la British Leyland, desarrolló un paquete de modelación por microordenador para diseñar sus nuevas factorías y cadenas de producción. El paquete, denominado "See why", desde entonces se ha puesto a la venta en Gran Bretaña. Aunque el mayor énfasis del programa recae en los procesos de modelación, incorpora salida gráfica en forma de diagramas esquemáticos. Los números dispuestos al lado de cada una de las etapas del proceso indican cómo avanza el trabajo y los problemas que surgen

Los puntos sobre las íes

El "generador de caracteres" es la sección de la memoria que define la forma en que aparecen los caracteres en la pantalla. En algunos sistemas el usuario puede diseñar sus propios símbolos

Anteriormente hemos visto en nuestro curso de programación BASIC (véase p. 214) que todos los caracteres alfanuméricos (y los símbolos gráficos, en el caso de que su ordenador los posea) se almacenan en la memoria RAM en forma de códigos de ocho bits (por lo general ASCII), de modo que un carácter ocupa un byte.

Cuando se imprime información en la pantalla, los códigos para cada carácter se colocan en una zona reservada de la memoria denominada *RAM de video*. Si, por ejemplo, se imprimiera la letra A en el rincón superior izquierdo de la pantalla, el primer byte de RAM de video contendría el código 65 (ASCII correspondiente a A). Si debajo de la A se imprimiera una C, y el ordenador tuviera una pantalla de 40 columnas, entonces en la 41 localización de RAM de video se hallaría el valor 67, y así sucesivamente. ¿Cómo convierte el ordenador el valor 65 en el patrón de puntos que componen en la pantalla el carácter A? La respuesta viene dada por un dispositivo denominado *generador de caracteres*.

Un generador de caracteres es sencillamente un conjunto de patrones almacenados en la memoria como bits. En los ordenadores personales el generador de caracteres está almacenado en ROM, lo que permite una visualización inmediata de los caracteres cuando se enciende la máquina. El generador de caracteres puede estar incorporado en las unidades ROM que contienen el intérprete de BASIC y el sistema operativo, o puede estar en un chip de ROM propio. En este último caso, con frecuencia el usuario encontrará proveedores independientes que le ofrezcan unidades ROM de recambio capaces de crear un juego de caracteres extranjeros, o una gama de símbolos especializados para, por ejemplo, ingeniería o matemáticas. Sin embargo, cada vez es mayor el número de máquinas que permiten transferir el generador de caracteres a RAM, gracias a lo cual el programador puede diseñar sus propios caracteres y símbolos.

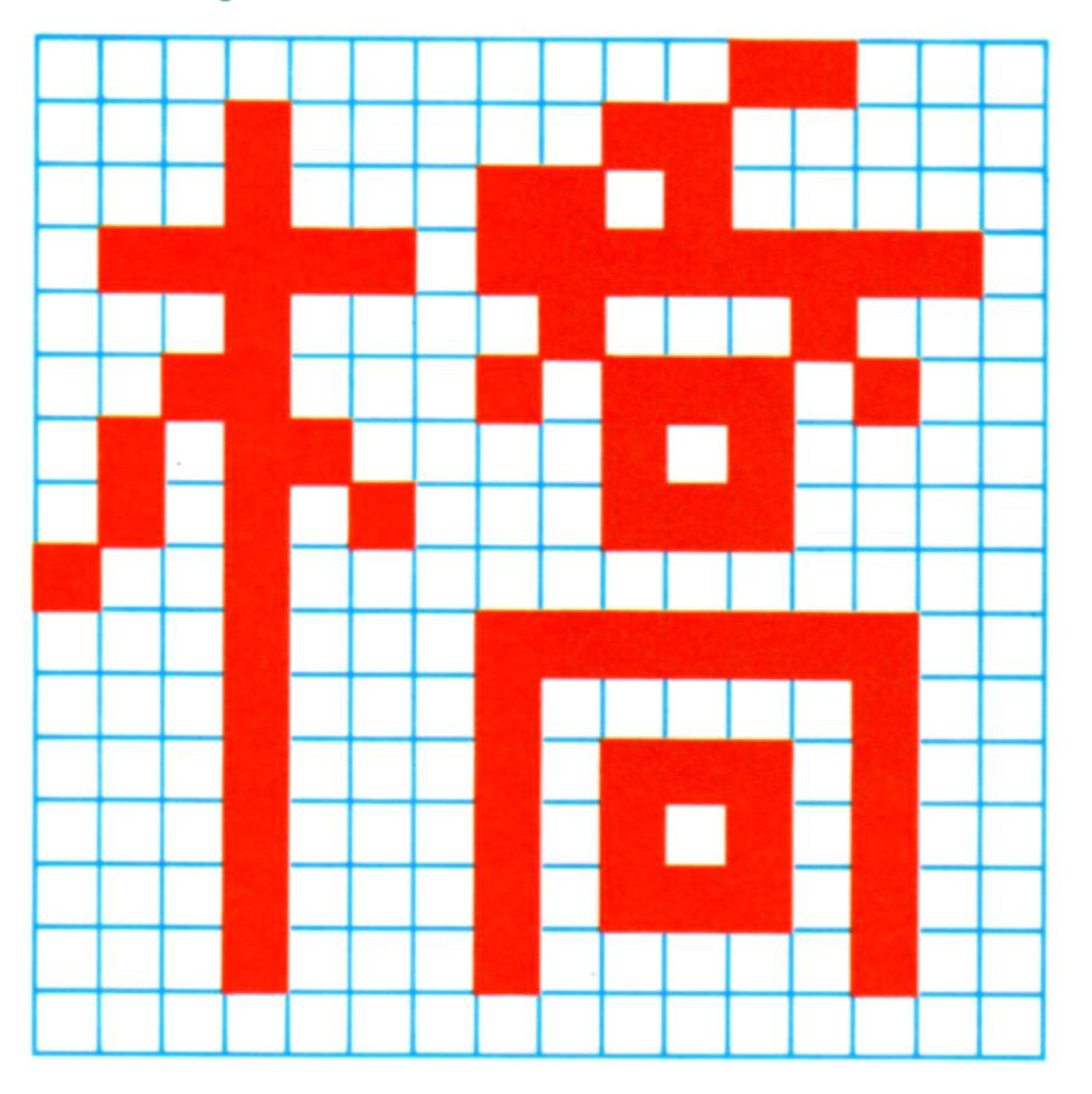
Todos los caracteres se construyen en una matriz de puntos, que en la mayoría de los ordenadores personales es de ocho por ocho, si bien con una matriz mayor se obtendría una mayor legibilidad y una gama más amplia de caracteres visualizables. Éstos se diseñan rellenando los cuadros de la cuadrícula. Se definen representando con un 1 los cuadrados ocupados y con un 0 los cuadrados en blanco, dando un total de 64 bits para cada carácter.

El primer byte del generador de caracteres representará el patrón de bits para la línea superior del primer carácter de la tabla. Si el ordenador sólo puede visualizar los caracteres ASCII con códigos de 0 a 127, el generador de caracteres requerirá entonces 128 × 8 bytes (1 Kbyte de memoria).

La dificultad para el ordenador reside en que cuando el barrido de la pantalla de televisión está generando la línea superior de la visualización, debe producir la línea superior de puntos para el carácter situado en el extremo superior izquierdo de la pantalla, seguido de la línea superior del carácter situado a su derecha, y así sucesivamente a través de la pantalla. Después, cuando el barrido empieza a pasar por segunda vez, debe hallar y visualizar la segunda línea para cada uno de los caracteres de la fila superior de la pantalla.

El sistema de circuitos de video consigue esto gracias a que posee dos contadores independientes. Uno lleva el registro de a qué localización de memoria de video corresponde el punto por el cual está pasando el barrido. El otro cuenta las líneas del barrido, empezando desde cero para la primera línea y llegando a siete para la octava, y volviendo luego a empezar desde cero en la novena, y así sucesivamente. De modo, entonces, que el ordenador busca el código ASCII o de visualización en la memoria de video, lo multiplica por ocho y suma el valor que en ese momento registra el contador de líneas. Esto le proporciona una dirección en el generador de caracteres para el patrón de ocho bits correspondiente a la fila correcta del carácter que se está barriendo en ese momento.

A modo de ejemplo, veamos cómo se generaría el carácter A, cuyo código ASCII es 65. Podemos calcular que la primera línea del carácter se almacenará en el byte número 520 (64 × 8 + 0); la segunda, en el byte 521 (65 × 8 + 1); la tercera línea, en el byte 522 (65 × 8 + 2); y así sucesivamente. Todo lo que falta es que el sistema de circuitos de video convierta esos ocho bits en una secuencia de voltajes que haga que el haz de electrones de barrido se encienda y se apague para visualizar el carácter en la pantalla.





Un carácter japonés
Los caracteres japoneses pueden ser sumamente complejos y la matriz normal de 8 × 8 no logra visualizarlos con suficiente detalle como para que resulten legibles. El carácter para "puente", que muestra la ilustración, apenas resulta legible en una matriz de 16 × 16. Utilizando una matriz de puntos de 24 × 24 se obtiene una mayor fidelidad

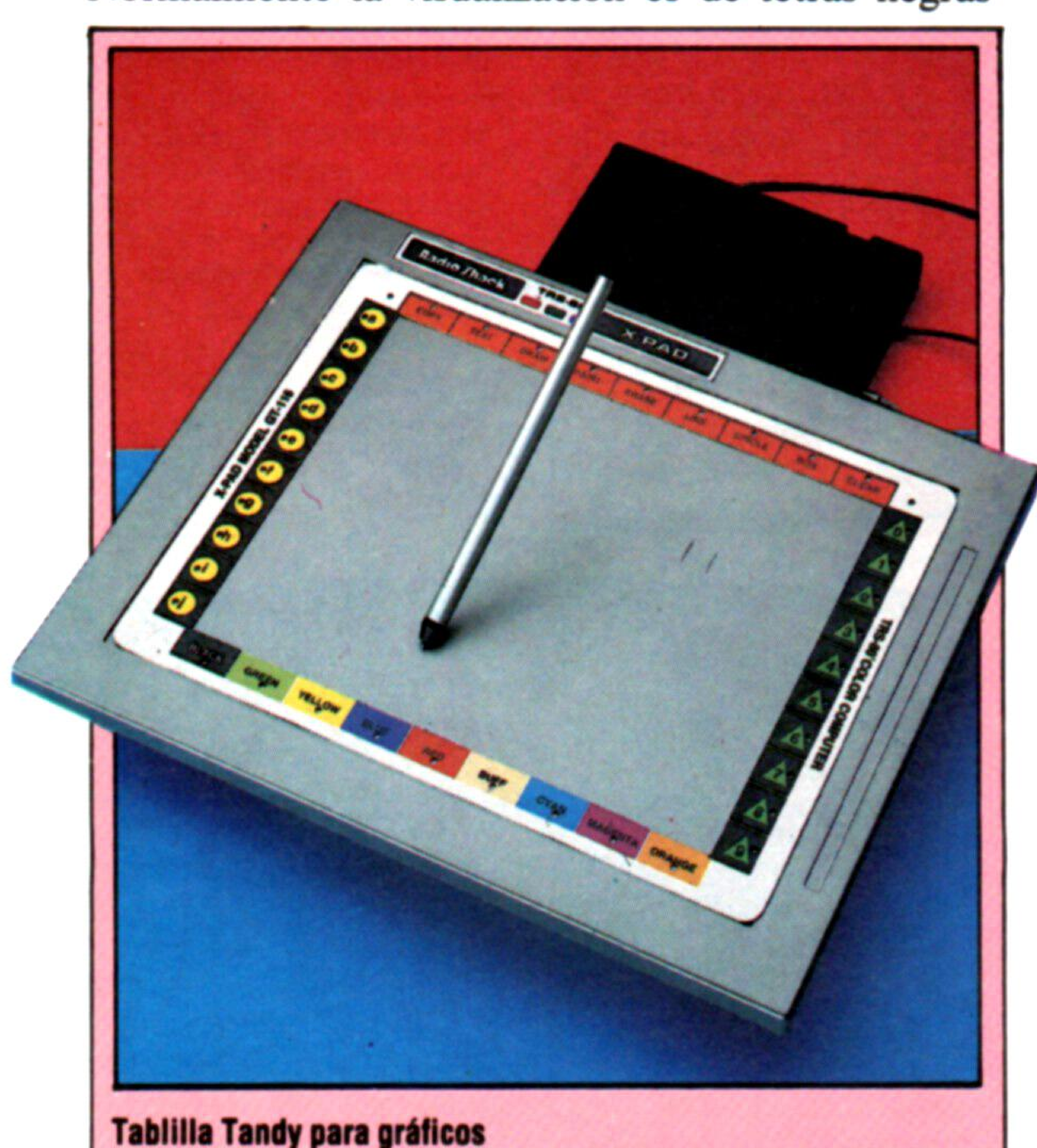
Tandy Color

De diseño muy similar al Dragon 32, este ordenador tiene un buen apoyo de periféricos: desde digitalizadores a impresoras de chorro de tinta

A pesar de ser bastante distinto en cuanto a su aspecto, el ordenador Tandy Color guarda notable parecido con el Dragon 32 (véase p. 130) por la forma en que funciona. De hecho, son tan compatibles que muchos de los programas que se ejecutan en uno pueden ejecutarse en el otro.

Existen, asimismo, otras semejanzas. Ambos poseen el mismo tipo de conexión para cartucho y la misma CPU: una 6809E. Ésta es una unidad de procesamiento muy potente, poco utilizada por otros ordenadores personales. Pero en algunos aspectos esta CPU está descompensada respecto a los otros componentes del ordenador. Normalmente, la 6809 se emplea en máquinas diseñadas para uso profesional. Sin embargo, en el Tandy Color, que posee una velocidad de reloj relativamente lenta (895 KHz), su potencial se desperdicia bastante.

El ordenador viene con su propia versión de BASIC: el BASIC de Tandy Color. A pesar de la poca velocidad del reloj, el funcionamiento del ordenador es aceptablemente rápido gracias a la sofisticación de la CPU. Posee varias palabras órdenes especializadas, diseñadas para que el hardware resulte más sencillo de utilizar. Una de las configuraciones del Tandy Color es el controlador de video 6847, un dispositivo programable que produce un formato de pantalla de 16 líneas de 32 caracteres. Normalmente la visualización es de letras negras



Tablilla Tandy para gráficos
La tablilla para gráficos GT-116 se puede utilizar ya sea como digitalizador, para trazar dibujos e imágenes en el ordenador, o bien como dispositivo señalador, para seleccionar ítems de los menús visualizados en la pantalla del ordenador

El teclado Tandy
A diferencia del Dragon, el Tandy Color no posee un teclado estilo máquina de escribir, sino que, en cambio, tiene 53 botones cuadrados. Éstos están suficientemente espaciados, pero no ofrecen la adecuada "sensación táctil" para largos períodos de escritura al tacto

Conectores para palanca de mando
Son totalmente proporcionales e incluyen una línea de señalización, que generalmente está conectada a un botón. Estas interfaces se pueden utilizar para otras señales analógicas, como las provenientes de experimentos de laboratorio

Interruptor

Componentes de la fuente de alimentación
Acondicionan y dan uniformidad a la salida del transformador. En el proceso se genera muchísimo calor, que es eliminado por los grandes disipadores

Transformador
La fuente de alimentación eléctrica incorporada consigue que el montaje sea más pulcro, con menos cables secundarios y cajas separadas

Conexión RS232
Aquí se pueden conectar impresoras, modems u otros dispositivos en serie. La máquina no tiene previstas entradas o salidas en paralelo

Interface para cinta
Esta interface proporciona también control remoto del motor

Selección de canal
La salida de TV se puede sintonizar en dos canales diferentes. La selección se efectúa cambiando la posición de este interruptor

ROM de 8 K
Contiene el BASIC de Tandy Color y rutinas de bajo nivel

Controlador de video 6847
El estilo y el color de la visualización en pantalla se controlan programando este chip, ya sea en BASIC o en código de lenguaje máquina

Conector para el teclado
A través de esta conexión, un cable plano de plástico laminado une el teclado con el ordenador





sobre un fondo verde, si bien tanto el color del fondo como el del primer plano se pueden escoger de entre nueve colores y se puede alterar el formato. Se proporcionan facilidades para generación de sonido, pero al disponer de sólo un canal y carecer de control de envoltura y generación de "ruido blanco", esta máquina es menos flexible que otras.

El Tandy Color posee el mismo tipo de carga de cartuchos ROM que el Dragon, y la gama de software disponible en este formato es muy amplia. Existe la variedad normal de cartuchos de juegos: ajedrez, cartas, damas, prospecciones petrolíferas y pinball. El software disponible incluye cartuchos de música, aritmética y mecanografía. También existe una buena gama de programas de utilidad para presupuestos, archivos y redacción de cartas, así como algunos medios auxiliares para desarrollo de programas para BASIC o código de lenguaje máquina. Además, existe una ROM de diagnóstico muy fácil de manejar, diseñada para asegurar el correcto funcionamiento de la máquina.

En la parte posterior, una interface en serie RS232 permite la utilización de una gama de dispositivos externos, incluyendo impresoras en serie. Esta conexión se puede usar también para comunicarse con otros ordenadores mediante un modem (véase p. 216). Con un dispositivo de este tipo y con el cartucho de videotex se pueden buscar diversas bases de datos o enviar correo electrónico.

También en la parte posterior de la máquina se proporcionan dos conectores para palanca de mando, cada uno de los cuales posee dos ejes de movimiento y un botón de disparo. Esto parece plantearle un pequeño problema al BASIC de Tandy Color: cuando se pulsa el botón se genera un flujo de caracteres. Las palancas de mando son mucho mejores que otras, con movimiento variable en lugar del tipo de mecanismo más usual y sencillo de empujar y tirar, y se pueden emplear para leer otras entradas analógicas (de valor variable), como las de los experimentos físicos.

La ampliación de la máquina se efectúa a través de la conexión para cartucho y se pueden conectar hasta cuatro unidades de disco de 5 ¼ pulgadas, que proporcionan un estimable almacenamiento total de 626 Kbytes.



Chris Stevens

Reloj de video
El tamaño y el número de caracteres en la pantalla se establecen por la frecuencia generada por este cristal, que también se utiliza para la producción real de la imagen

Interruptor de borrado
Pulsándolo limpiará la máquina y ésta empezará de nuevo

Carga de cartuchos
Se utiliza para conectar unidades de disco y software operativo, además de cartuchos corrientes de programas y juegos

RAM
Esta máquina posee 16 K de RAM dinámica en 8 chips, cada uno de los cuales abarca un bit del byte de 8 bits

CPU 6809E
Un chip avanzado con aritmética de 16 bits interna, que opera sobre un bus de datos de 8 bits y puede direccionar 64 K de memoria

Enchufe para ROM de 8 K de recambio
Se puede emplear para servicios adicionales o ampliar el BASIC

TANDY COLOR

DIMENSIONES
369 x 344 x 94 mm

VELOCIDAD DEL RELOJ
0,895 MHz

MEMORIA
4 Kbytes de RAM; 8 Kbytes de ROM, ampliables a 32 Kbytes

VISUALIZACION EN VIDEO
16 líneas de 32 caracteres. Nueve colores con selección independiente de primer plano y fondo. 127 caracteres predefinidos y 127 caracteres definibles por el usuario

INTERFACES
Conexión RS232, cassette, dos palancas de mando proporcionales, interface para TV en dos canales

LENGUAJE SUMINISTRADO
BASIC

OTROS LENGUAJES DISPONIBLES
Código de lenguaje máquina 6809 con paquetes Assembler

VIENE CON
Manuales de instalación y de BASIC, cable para TV

TECLADO
53 teclas individuales, dispuestas como en una máquina de escribir

DOCUMENTACION
La presentación del manual del usuario es desigual, pero éste es muy amplio y constituye una guía perfectamente adecuada para la máquina



Las impresoras Tandy
El Tandy Color está bien apoyado por una gama de impresoras que comercializa la propia casa Tandy. La CGP-220, por ejemplo, utiliza el principio del chorro de tinta para producir siete colores, y su funcionamiento es silencioso. La CGP-115, más barata que la anterior, utiliza cuatro lápices esferográficos en miniatura para producir una impresión en color y trazar gráficos

Chris Stevens

Nuevas entradas

Para poder insertar una nueva entrada en una matriz, en primer lugar es necesario hallar un espacio vacío. La búsqueda binaria es una forma muy eficaz de lograrlo

En el capítulo anterior vimos que un archivo de datos se compone de registros, que a su vez se dividen en campos, a cada uno de los cuales se le puede dar acceso a los otros campos mediante un campo de clasificación. Ahora examinaremos algunas de las técnicas para buscar en estas listas.

Crear registros para nuestra agenda de direcciones no es difícil. Supongamos que existe una matriz en serie separada para cada uno de los campos de registros. Éstas se podrían denominar NOMBRES$ (para nombre completo) CALLE$, CIUDAD$ y TELEFONO$ (luego hablaremos de por qué hemos utilizado aquí una variable alfanumérica para el campo del número de teléfono en vez de una variable numérica). En la lista de las ocho funciones deseables del programa de la agenda de direcciones, el número seis era la capacidad de agregar nuevas entradas. Si estas ocho opciones se presentaran en la pantalla al principio, cuando se ejecutara el programa, al seleccionar 6 lo llevaría a una rutina de entrada del tipo que presentamos como ejercicio.

Supongamos que en la agenda ya hay cierto número de entradas, pero que no recuerda cuántas. Es esencial que las nuevas entradas no se escriban sobre las entradas existentes; de este modo una de las tareas del programa podría consistir en buscar a través de los elementos de una de las matrices para hallar la primera que no contenga datos.

Buscar en una matriz para ver si un elemento está "ocupado" no es difícil. En BASIC las variables alfanuméricas se pueden comparar del mismo modo que se comparan las variables numéricas. IF A$ = "COMPUTER" THEN... es tan válido como IF A = 61 THEN..., al menos en la mayoría de las versiones de BASIC. Si cualquiera de las matrices de nuestra agenda de direcciones ya posee una entrada, ésta constará de al menos un carácter alfanumérico. Un elemento "vacío" no contendrá ningún carácter alfanumérico, de manera que todo lo que tenemos que hacer es buscar a través de los elementos, empezando desde el principio, hasta que hallemos uno que no contenga caracteres.

Si hay matrices para el nombre, la calle, la ciudad y el número de teléfono, tendremos cuatro matrices con un elemento en cada una para cada campo del registro. Puesto que todos estos campos "van juntos", el registro 15 tendrá sus datos de nombre en el elemento 15 de la matriz de nombre, sus datos de calle en el elemento 15 de la matriz de calle, los datos de ciudad en el elemento 15 de la matriz de ciudad, y los datos del número de teléfono en el elemento 15 de la matriz de número de teléfono. Por lo tanto, sólo necesitamos buscar en una de estas matrices para hallar un elemento vacío; no necesitamos revisar todas las matrices.

Si la variable POSICION representa el número del primer elemento libre de cualquiera de las matrices, un programa para localizar POSICION (en el supuesto de que no lo sepamos ya) podría ser tan sencillo como el siguiente:

```
PROCEDIMIENTO (hallar elemento libre)
BEGIN
  LOOP
    REPEAT UNTIL localizar elemento libre
    READ matriz (POSICION)
    POSICION = POSICION + 1
    IF matriz (POSICION) = " "
      THEN tomar nota POSICION
      ELSE no hacer nada
    ENDIF
  ENDLOOP
END
```

En BASIC, esto tendría un desarrollo sencillo:

```
1000 FOR L= 0 TO 1 STEP 0
1010 LET POSICION = POSICION + 1
1020 IF NOMBRES$ (POSICION) = " " THEN
     LET L = X
1030 NEXT L
1040 REM resto del programa
```

Observe que en la línea 1020 el valor de X es el que se requiere para terminar el bucle FOR...NEXT y este valor varía de máquina a máquina (véase el recuadro "Complementos al BASIC"). También es importante observar que éste es un fragmento de programa y se supone que NOMBRES$ ( ) está DIMensionada y que se ha inicializado POSICION. Para ejecutar este fragmento como si se tratara de un programa, debe DIMensionar NOMBRES$ ( ) e inicializar POSICION y X en algún momento antes de la línea 1000.

Aunque ya hemos utilizado anteriormente la técnica FOR X = 0 TO 1 STEP 0, éste es un buen momento para analizar con más detalle la forma en que funciona. Por lo general, un bucle FOR...NEXT en BASIC "sabe" de antemano cuántas veces se espera que se repita el fragmento de programa. Si desea repetir algo 30 veces, FOR X = 1 TO 30 lo conseguirá de forma precisa. No obstante, esta vez estamos simulando un bucle REPEAT...UNTIL (repetir hasta). Aunque las versiones corrientes de BASIC no disponen de REPEAT...UNTIL, simularlo mediante un FOR...NEXT es bastante fácil. En la medida en que fracase la condición de la línea 1020, L (el contador del bucle FOR...NEXT) permanece en el valor 0, sumándosele 0 a cada iteración (repetición del bucle), mientras que la línea 1010 hace que POSICION se incremente en 1 a cada iteración. Cuando la condición de la línea 1020 es verdadera (o sea, cuando se ha hallado un elemento de NOMBRES$( ) vacío), L se establece en el valor X, y el bucle FOR...NEXT acaba en la línea 1030. Con ello POSICION señala el primer elemento libre de NOMBRES$( ).

POSICION es un valor que posiblemente necesitemos establecer antes, cada vez que se emplea el programa de la agencia de direcciones, y es un valor que es probable sea necesario actualizar varias veces durante la utilización del programa. Por lo tanto será una de nuestras variables "globales" y será necesario que el establecimiento de su valor forme parte de una rutina de "inicialización". Esto se puede hacer cada vez que se ejecuta el programa, o se puede crear una "bandera" si el valor de POSICION ha cambiado o no desde la última vez que se ejecutara el programa. Este último enfoque no es difícil, pero en este punto viene a crear una complicación innecesaria. Nosotros dejaremos que las cosas sigan siendo sencillas y hallaremos el valor de POSICION como una de las primeras tareas cada vez que se ejecuta el programa.

Pasemos revista a las actividades que deseamos que haga la agenda de direcciones computerizada y veamos si podemos avanzar hacia una estrategia de programa total. Esta vez seremos un poco más rigurosos y daremos por sentado que cada una de las actividades se tratará como subrutina separada (el nombre de cada una se indicará entre asteriscos).

1. Hallar registro (del nombre) *ENCREG*
2. Hallar nombres (de nombres incompletos) *ENCNOMBRE*
3. Hallar registro (de ciudad) *ENCCIUDAD*
4. Hallar registros (de inicial) *ENCINICI*
5. Listar registros (todos) *LISTREGS*
6. Agregar registro *INCLREG*
7. Modificar registro *MODREG*
8. Borrar registro *BORREG*
9. Salida del programa (guardarlo) *SALPROG*

Ahora sabemos, en líneas generales, cuáles son las "entradas" y "salidas" deseadas del programa, de modo que ya podemos empezar a pensar en términos de un programa principal. Toda la pormenorización se puede efectuar a través del proceso de programación *top-down* (de arriba abajo), y se puede codificar en las diversas subrutinas. Sabemos que se deberán inicializar varias cosas, incluyendo el valor de POSICION. Sabemos que, como será un programa activado por menú, se nos presentará una serie de opciones cada vez que se ejecute el programa. Sabemos, asimismo, que independientemente de cuál sea nuestra respuesta a las opciones presentadas, desearemos que se ejecute al menos una de ellas. De manera que ya puede tomar forma el cuerpo del programa principal:

```
PROGRAMA PRINCIPAL
EMPEZAR
  INICIALIZACION (procedimiento)
  PRESENTACION (procedimiento)
  ELECCION (procedimiento)
  EJECUCION (procedimiento)
FIN
```

En BASIC, esto tendría el aspecto siguiente (con los números de línea reemplazados por los nombres de las subrutinas):

```
10 REM PROGRAMA AGENDA DE MI COMPUTER
20 GOSUB *INICIALIZACION*
30 GOSUB *PRESENTACION*
40 GOSUB *ELECCION*
50 GOSUB *EJECUCION*
60 END
```

La subrutina o procedimiento *PRESENTACION* visualizaría en la pantalla durante algunos segundos un saludo, seguido del menú. El saludo podría ser:

```
*BIEN VENIDO A LA*
*AGENDA COMPUTERIZADA*
*DE MI COMPUTER*
(PULSE LA BARRA ESPACIADORA CUANDO ESTE
LISTO PARA CONTINUAR)
```

En respuesta a la invitación a pulsar la barra espaciadora, el programa se bifurcará hacia la subrutina *ELECCION* y se le ofrecerá al usuario una pantalla como ésta:

```
*DESEA USTED*
1. HALLAR UN REGISTRO (de un nombre)
2. HALLAR NOMBRES (de parte de un nombre)
3. HALLAR REGISTROS (de una ciudad)
4. HALLAR REGISTROS (de una inicial)
5. LISTAR TODOS LOS REGISTROS
6. AGREGAR UN REGISTRO
7. MODIFICAR UN REGISTRO
8. BORRAR UN REGISTRO
9. SALIR Y GUARDAR
*ELIJA DE 1 A 9*
*SEGUIDO DE RETORNO*
```

En este punto, el programa se bifurcará hacia la subrutina apropiada, según el número al que se dé entrada. Ahora la estructura del programa está empezando a tomar forma. Es necesario que todas las opciones, a excepción de la número 9 (SALIR y GUARDAR), terminen con una instrucción para retornar a la subrutina *ELECCION*. Pero no hemos tenido en cuenta muchos detalles relativos a la organización interna de los datos. De ellos nos ocuparemos más adelante.

Supongamos que estamos ejecutando el programa, que éste ya posee todos los registros que necesitamos y que deseamos buscar un registro completo dando entrada solamente a un nombre. Para ello se requiere la opción 1: HALLAR UN REGISTRO (*ENCREG*). Antes de que intentemos diseñar esta parte del programa, consideremos algunos de los problemas que plantean las rutinas de búsqueda computerizada.

La búsqueda

Los libros de texto acerca de las técnicas de programación tienden a tratar conjuntamente la búsqueda y la clasificación. Los lectores recordarán que nosotros ya hemos abordado el tema de la clasificación en un programa diseñado para ordenar nombres alfabéticamente (véase p. 134). Tanto la clasificación como la búsqueda plantean puntos interesantes acerca de cómo se organizan los datos en un ordenador o en cualquier otro sistema de información.

Si una agenda de direcciones "manual" consistiera en una agenda de notas sin índice alfabético, y si las entradas se fuesen agregando a medida que fuera necesario, sin clasificarlas por orden alfabético, tendríamos una estructura de datos de las que se denominan "pilas". Una pila es un conjunto de datos agrupados por el orden en que llegan. Es obvio que una pila es la forma menos eficaz de organizar los datos. Cada vez que se desee hallar la dirección y el número de teléfono de alguien habrá que mirar a través de toda la agenda de direcciones. Lo mismo suele ocurrir con los sistemas informáticos, aunque existen ocasiones en las que los crite-

rios en virtud de los cuales se accede a los datos son tan impredecibles que una pila puede ser una estructura de datos tan buena como cualquier otra.

Una estructura de datos más organizada, y cuya utilización resulta mucho más sencilla tanto para las personas como para los ordenadores, es la que se consigue cuando la información se organiza de acuerdo con un sistema simple y reconocido. Una guía telefónica constituye un buen ejemplo de un conjunto de datos (nombres, direcciones y números de teléfono) donde el campo del nombre está ordenado de acuerdo con las sencillas reglas de la sucesión alfabética. En sí mismos, los números en el fondo están dispuestos al azar, pero los nombres (que son más "significativos") están ordenados según unas reglas muy fáciles de seguir.

De acuerdo con la organización interna de los datos de nuestra agenda de direcciones computerizada, los datos están dispuestos en una pila, almacenándose un registro en el elemento X de la matriz de "nombre", en el elemento X de la matriz de "calle", etc., y almacenándose el registro siguiente en el elemento X +1 de la matriz de "nombre", en el elemento X +1 de la matriz de "calle", y así sucesivamente. Hallar un dato determinado (JUAN FLORES, por ejemplo) implicaría, por lo tanto, mirar el primer elemento de la matriz de "nombre" y ver si corresponde a JUAN FLORES, mirar el segundo elemento y ver si se refiere a JUAN FLORES, y así sucesivamente hasta que se logre localizar el campo o bien, por el contrario, descubrir que no hay ninguna entrada para JUAN FLORES.

Si el dato que deseamos buscar ya se hubiese ordenado en una estructura reconocible, veríamos cuánto se simplificaría la búsqueda. Supongamos que posee una base de datos sobre equipos de fútbol y que uno de los campos de los registros es el de los resultados de una semana determinada. Una base de datos eficaz le permitiría hallar qué equipo o equipos marcaron 11 goles durante esa semana. Ésta sería la matriz que retendría los marcadores de los equipos para la semana en cuestión:

1,6,2,2,1,9,0,0,2,1,4,11,4,2,12,5,2,1,0,1

Debería resultar obvio que los marcadores están dispuestos por orden de los equipos y no por el orden de los marcadores. Son veinte los equipos que participaron, y sólo uno consigue marcar 11 goles esa semana. Éste fue el 12.º equipo al que se dio entrada en la matriz. Con unos datos no estructurados como éstos, la única forma de hallar la información que se desea consiste en mirar el primer elemento y ver si es 11; de no serlo, mirar el elemento siguiente para comprobar si es 11, y así sucesivamente hasta localizar un 11 o descubrir que no hay ningún elemento cuyo valor sea igual a 11.

Si analizáramos estos datos, veríamos que había un total de 20 marcadores, cuyos valores oscilaban entre 0 y 12. Este ejemplo es relativamente trivial, e incluso aunque tuviéramos que buscar a través de cada dato no nos llevaría mucho tiempo descubrir que el 11 estaba en el 12.º elemento de la matriz. Pero, ¿y si en una gran matriz hubiera miles de elementos? Buscar entre una cantidad de datos muy numerosa retrasaría de manera considerable un programa.

La solución consiste en ordenar primero los datos, de modo que se pueda efectuar la búsqueda con mucha más rapidez. A continuación ofrecemos nuevamente la matriz de marcadores, dispuestos en orden numérico:

0,0,0,1,1,1,1,1,2,2,2,2,2,4,4,5,6,9,11,12

Si sabemos que el número de equipos es 20, entonces la forma más rápida de hallar la posición del marcador que deseamos consiste en dividir la matriz en dos partes y buscar sólo en la que probablemente contendrá el número que deseamos. Recuerde que es probable que examinar grandes cantidades de datos lleve mucho más tiempo que operaciones aritméticas tan sencillas como dividir un número por dos. El algoritmo para localizar el marcador tendría ahora la siguiente configuración:

```
Hallar la matriz que contenga los marcadores
Leer el número que deseamos buscar
Hallar la longitud de la matriz
Hallar el punto medio de la matriz
Hacer un bucle hasta localizar el número
  Si el dato situado en el punto medio es igual al
  número que estamos buscando, entonces se ha
  localizado el número
  Si no lo es, ver si el número buscado es mayor
  o menor que el situado en el punto medio
  Si el número buscado es mayor que el número
  situado en el punto medio, hallar el punto
  medio de la parte superior de la matriz
  Si el número requerido es menor que el número
  situado en el punto medio, hallar el punto
  medio de la parte inferior de la matriz
  (Repetir este procedimiento hasta localizar el
  número)
```

A esto le podríamos dar la siguiente forma:

```
EMPEZAR
  Hallar la matriz de marcadores
  INPUT NUMERO (a buscar)
  LOOP hasta localizar el número
    IF NUMERO = (punto medio)
      THEN apuntar posición punto medio
      ELSE
        IF NUMERO > (punto medio)
          THEN hallar punto medio de mitad
            superior
          ELSE hallar punto medio de mitad
            inferior
        ENDIF
      ENDIF
  ENDLOOP
  IF NUMERO está localizado
    THEN PRINT posición de punto medio
    ELSE PRINT "NUMERO NO HALLADO"
  ENDIF
END
```

Si piensa en este programa escrito en seudolenguaje, verá que finalmente no puede fracasar en localizar el número que se está buscando si existe en la matriz. Desarrollemos este seudolenguaje hasta llegar a un programa de trabajo. Este proceso de búsqueda mediante subdivisión repetida se denomina *búsqueda binaria*.

Le ofrecemos, para que pruebe con él, un programa en BASIC fundamentado en el seudolenguaje anterior. Crea una matriz y lee los marcadores desde una sentencia de datos. Luego se prepara para buscar el marcador. Si lo encuentra, imprime el elemento de la matriz en el que se encontró el número.

```
10 REM UN PROGRAMA PARA LOCALIZAR UN
   NUMERO EN UNA MATRIZ
20 DIM MARCADORES (20)
30 FOR Z = 1 TO 20
40 READ MARCADORES (Z)
50 NEXT Z
60 DATA 0,0,0,1,1,1,1,1,2,2,2,2,2,4,4,5,6,9,11,12
70 LET L = 20
80 LET BTM = 1
90 LET TP = L
100 INPUT "DE ENTRADA AL MARCADOR";N
110 FOR Z = 0 TO 1 STEP 0
120 LET L = TP - BTM
130 LET MD = BTM + INT(L/2)
140 IF N = MARCADORES(MD) THEN LET Z = X
150 IF N > MARCADORES(MD) THEN LET
    BTM = MD
160 IF N < MARCADORES(MD) THEN LET
    TP = MD
170 NEXT Z
180 PRINT "EL MARCADOR ESTABA EN
    EL ELEMENTO N.° ";MD
190 END
```

Nuevamente, observe que se habrá de inicializar X según las exigencias de su máquina (véase el recuadro "Complementos al BASIC").

Si los datos retenidos en un archivo o en una matriz son bastante regulares, como en el caso de una guía telefónica, donde los nombres están distribuidos con razonable uniformidad a través del alfabeto, entonces la búsqueda binaria es un procedimiento eficaz para hallar una entrada determinada. No obstante, no es de ningún modo la forma más eficaz, y hay algoritmos alternativos que pueden encontrar los datos utilizando menos iteraciones. Uno de ellos es la técnica *hashing*, en que el programa hace una conjetura sobre la localización de la entrada, y la va refinando hasta hallarla. Pero esto rebasa los límites de este curso, y el proceso de búsqueda binaria basta para nuestras necesidades.

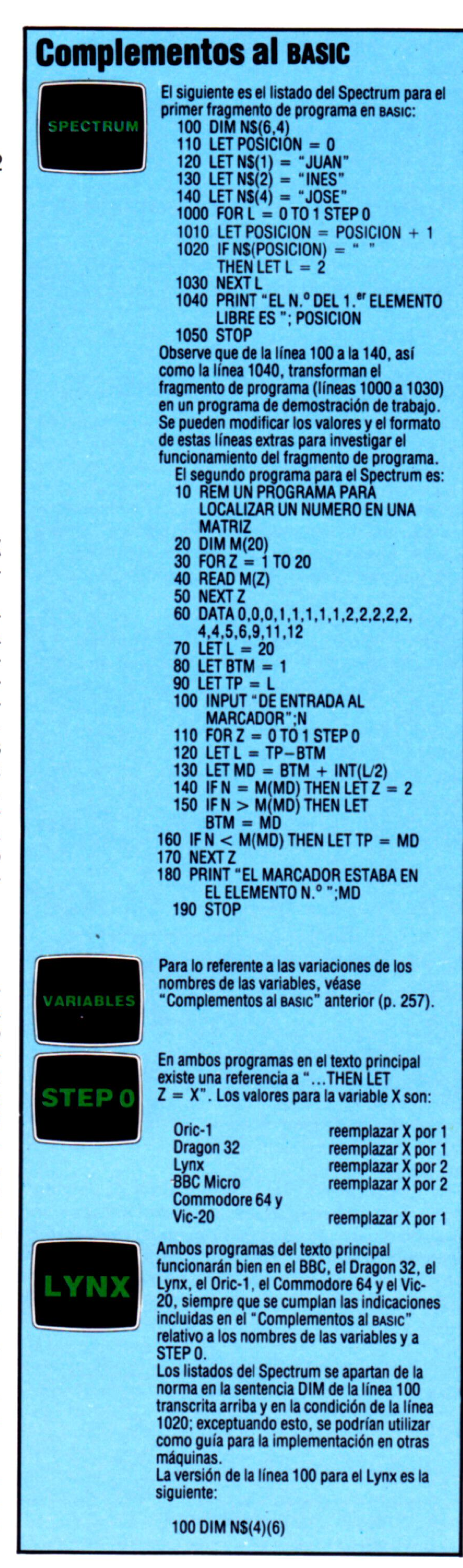
Ejercicios

Si ejecuta este programa, verá que funciona siempre y cuando dé entrada a un marcador que exista en la matriz. Si diera entrada a un marcador como 3, que no está en la matriz, el programa no conseguiría llegar hasta el final y no aparecería ningún mensaje de error. Si digitara 12, que sí se halla en la matriz, el programa fracasaría en localizarlo. El programa también da por sentado que todos los números de la matriz clasificada serán diferentes, pero, como puede observar a partir de la sentencia de datos, varios números se dan más de una vez. El programa no detecta esto ni informa de todas las localizaciones donde se produce el número.

Su tarea consiste en:

1. Analizar el programa y descubrir por qué éste no puede localizar un marcador de 12
2. Modificar una línea del programa para rectificar este defecto
3. Descubrir por qué el programa no puede manipular números que no existan en la variable e idear una estrategia para superar el problema

En la página 235 le ofrecíamos una serie de ejercicios de revisión para ayudarlo a valorar su nivel de aprovechamiento en nuestro curso de programación BASIC. Hallará las soluciones en la página 280.



Complementos al BASIC

SPECTRUM

El siguiente es el listado del Spectrum para el primer fragmento de programa en BASIC:

```
100 DIM N$(6,4)
110 LET POSICION = 0
120 LET N$(1) = "JUAN"
130 LET N$(2) = "INES"
140 LET N$(4) = "JOSE"
1000 FOR L = 0 TO 1 STEP 0
1010 LET POSICION = POSICION + 1
1020 IF N$(POSICION) = " "
     THEN LET L = 2
1030 NEXT L
1040 PRINT "EL N.° DEL 1.er ELEMENTO
     LIBRE ES "; POSICION
1050 STOP
```

Observe que de la línea 100 a la 140, así como la línea 1040, transforman el fragmento de programa (líneas 1000 a 1030) en un programa de demostración de trabajo. Se pueden modificar los valores y el formato de estas líneas extras para investigar el funcionamiento del fragmento de programa.

El segundo programa para el Spectrum es:

```
10 REM UN PROGRAMA PARA
   LOCALIZAR UN NUMERO EN UNA
   MATRIZ
20 DIM M(20)
30 FOR Z = 1 TO 20
40 READ M(Z)
50 NEXT Z
60 DATA 0,0,0,1,1,1,1,1,2,2,2,2,2,
   4,4,5,6,9,11,12
70 LET L = 20
80 LET BTM = 1
90 LET TP = L
100 INPUT "DE ENTRADA AL
    MARCADOR";N
110 FOR Z = 0 TO 1 STEP 0
120 LET L = TP-BTM
130 LET MD = BTM + INT(L/2)
140 IF N = M(MD) THEN LET Z = 2
150 IF N > M(MD) THEN LET
    BTM = MD
160 IF N < M(MD) THEN LET TP = MD
170 NEXT Z
180 PRINT "EL MARCADOR ESTABA EN
    EL ELEMENTO N.° ";MD
190 STOP
```

VARIABLES

Para lo referente a las variaciones de los nombres de las variables, véase "Complementos al BASIC" anterior (p. 257).

STEP 0

En ambos programas en el texto principal existe una referencia a "...THEN LET Z = X". Los valores para la variable X son:

Oric-1	reemplazar X por 1
Dragon 32	reemplazar X por 1
Lynx	reemplazar X por 2
BBC Micro	reemplazar X por 2
Commodore 64 y Vic-20	reemplazar X por 1

LYNX

Ambos programas del texto principal funcionarán bien en el BBC, el Dragon 32, el Lynx, el Oric-1, el Commodore 64 y el Vic-20, siempre que se cumplan las indicaciones incluidas en el "Complementos al BASIC" relativo a los nombres de las variables y a STEP 0.

Los listados del Spectrum se apartan de la norma en la sentencia DIM de la línea 100 transcrita arriba y en la condición de la línea 1020; exceptuando esto, se podrían utilizar como guía para la implementación en otras máquinas.

La versión de la línea 100 para el Lynx es la siguiente:

```
100 DIM N$(4)(6)
```

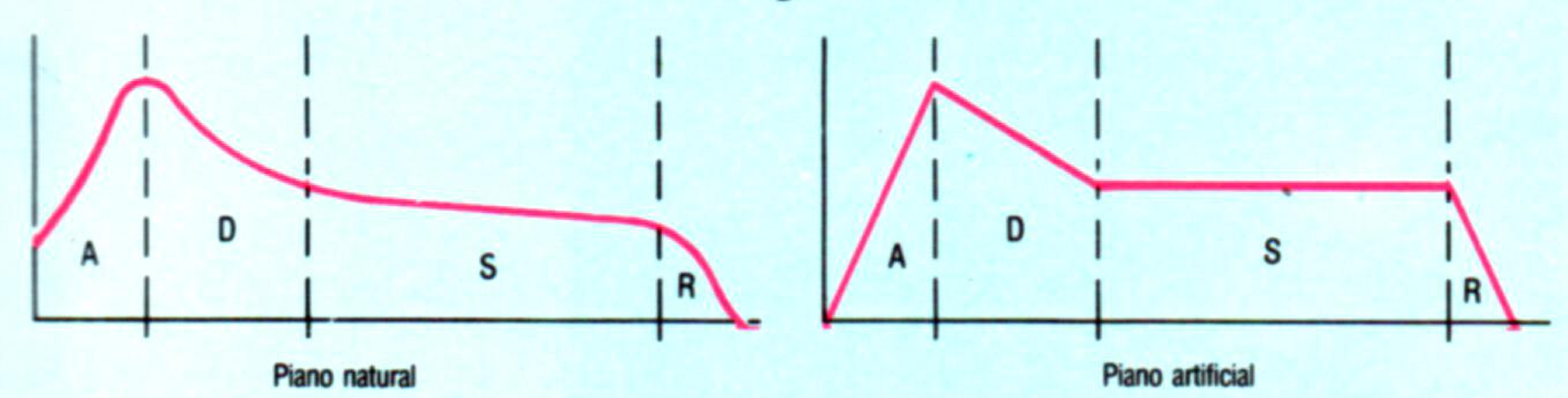
Sonidos fieles

Continuamos con el tema de la música por ordenador

Como parte del estudio acerca de la generación de sonido con un microordenador, veremos ahora algunas de las configuraciones más avanzadas.

Generadores de envoltura

La *envoltura* de un sonido es el patrón de sus cambios de volumen, desde que se ejecuta hasta que se extingue. El diagrama ilustra envolventes similares a las de una nota de piano y de otros instrumentos. Las envolventes se dividen en cuatro elementos, que se suelen denominar ADSR (*Attack, Decay, Sustain, Release*: arranque, demora, sostenido y liberación). En el caso del piano, el volumen se eleva rápidamente hasta su nivel más alto una vez se pulsa la tecla (arranque), luego disminuye más lentamente (demora) hasta el nivel de volumen más constante (sostenido) mientras la tecla se mantiene oprimida, y por último cae rápidamente a cero (liberación) cuando se suelta la tecla. El sostenido es un nivel de volumen, mientras que el arranque, la demora y la liberación son intervalos de tiempo. El dispositivo capaz de controlar estos cuatro aspectos de una envoltura se denomina *generador ADSR*. Todo dispositivo capaz de encender y apagar un sonido es un generador de envoltura de clases.

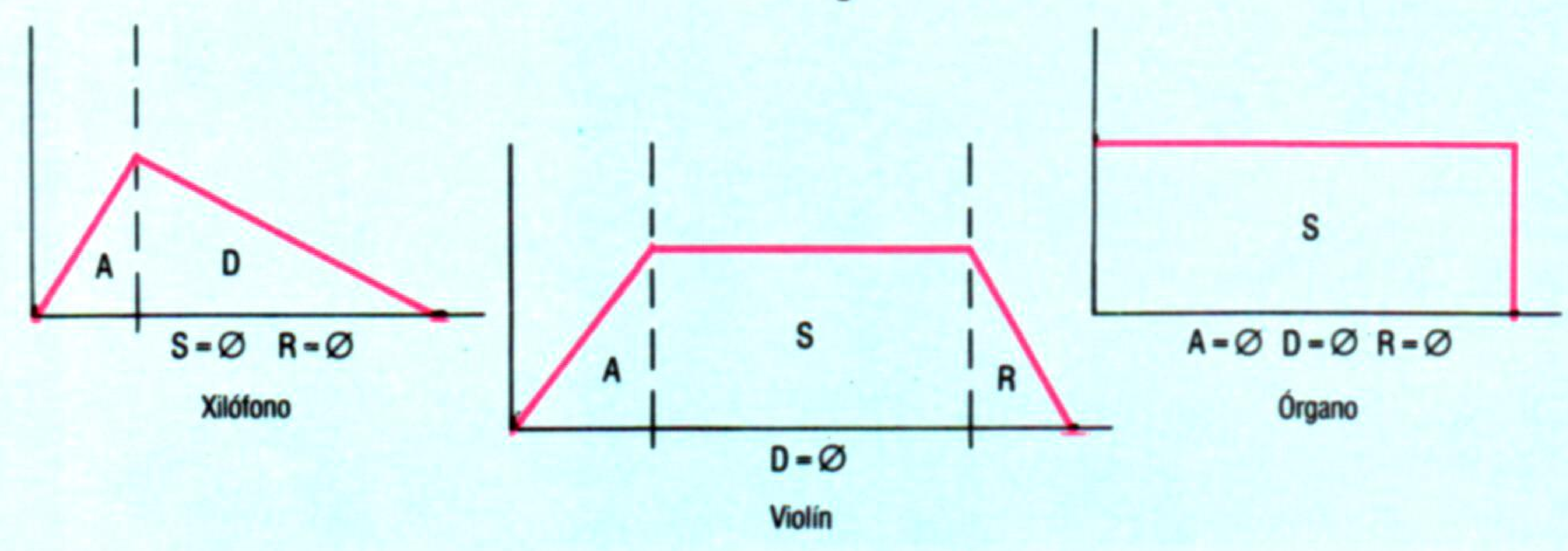


Piano natural

Piano artificial

La mayoría de los ordenadores no posee otra sofisticación más que un generador *beep*, que no es más que un interruptor para encender el sonido a un volumen constante durante un tiempo específico. En este caso los elementos A, D y R son iguales a cero.

El Oric-1 proporciona cierto control de envoltura, pero en este sentido los ordenadores más versátiles son el BBC Micro y el Commodore 64; ambos poseen generadores ADSR, capaces de imitar las envolventes de los instrumentos más comunes, así como las de algunos instrumentos "artificiales".



Xilófono

Violín

Órgano

...Dibujos atrayentes

Animación sencilla utilizando las órdenes POKE y PRINT

Habiendo analizado los principios básicos de los gráficos por ordenador, ahora vamos a ver cómo se puede obtener una animación sencilla. Primero, sin embargo, será necesario detallar las formas en que se pueden hacer aparecer los caracteres en la pantalla y cómo se pueden controlar sus posiciones empleando un programa en lenguaje BASIC. El programador dispone de dos métodos principales: las órdenes POKE y PRINT, con sus correspondientes funciones.

La orden POKE coloca un número en cualquier posición de memoria especificada. En el interior de todo ordenador existe una serie especial de posiciones de memoria, cada una de las cuales está relacionada con una posición específica del carácter en la pantalla. En una pantalla estándar de 25 filas por 40 columnas, se reservan 1 000 posiciones con esta finalidad. Cada posición retiene un número que corresponde a un carácter determinado de ese juego de caracteres de la máquina. Éste puede ser el código ASCII del carácter (véase p. 214) o un código diseñado por el fabricante de la máquina. Además de estas posiciones codificadas de caracteres, por lo general existe otro juego de posiciones que retienen información relativa al color de un carácter visualizado en cualquier lugar de la pantalla.

En el Commodore 64, por ejemplo, existen muy pocas órdenes de gráficos en BASIC para ayudar al programador, y a menudo se utiliza la orden POKE para crear visualizaciones en pantalla. Las direcciones de las posiciones que retienen códigos de caracteres van de 1024 a 2023, y las posiciones que retienen la información de color poseen direcciones entre 55296 y 56295. Para el Commodore, el carácter A posee un código de pantalla 1 y el color negro se representa por un código de color 0; en consecuencia, las órdenes que se requieren para colocar una letra A negra en el ángulo superior izquierdo de la pantalla son:

```
10 POKE 1024,1
20 POKE 55296,0
30 END
```

Se pueden realizar modificaciones sencillas para visualizar una fila de letras A negras en la línea superior de la pantalla:

```
10 FOR X = 0 TO 39
20 POKE 1024 + X,1
30 POKE 55296 + X,0
40 NEXT X
50 END
```

Las letras A las produce el bucle FOR...NEXT, que cada vez incrementa en uno las direcciones de la posición del carácter y del color. Si incorporáramos una orden para borrar una A antigua cada vez que se creara una nueva, entonces parecería que la A se

moviera a través de la pantalla: ¡una forma no muy refinada de animación!

El código de caracteres del Commodore para un espacio es 32. Todo lo que se necesita es que el programa lo coloque en la posición adecuada, un lugar detrás de donde se va a visualizar la nueva A. Inserte esta línea en el programa anterior:

```
35 POKE 1024 + X,32
```

O, como alternativa:

```
15 POKE 1024 + (X – 1),32
```

Utilizar este sistema para producir gráficos es un proceso laborioso. Probablemente lo más adecuado sea crear visualizaciones de fondo estático empleando sentencias READ y DATA para dar entrada a los códigos de los caracteres antes de almacenarlos (POKE) en la posición correcta.

La mayoría de los ordenadores personales posee muchas órdenes para gráficos como parte de su juego de instrucciones en BASIC estándar, que permiten que el usuario cree notables visualizaciones, plenas de color, utilizando sólo unas pocas sentencias simples. A menudo se incluyen órdenes para construir patrones geométricos de alta resolución. Con estas instrucciones el usuario puede trazar puntos en la pantalla y unirlos entre sí por medio de líneas rectas; puede dibujar cuadrados, arcos y círculos y colorear los interiores de las formas dibujadas.

Un método muy directo suele ser el de volver a trazar formas antiguas con el mismo color que el fondo, lo que equivale a borrarlas. Trazar rápidamente, borrar y volver a trazar en una nueva posición es, de hecho, la base de la animación gráfica sencilla. El realismo de la acción depende en gran medida de la velocidad con que se pueda llevar a cabo el proceso. Los sprites son mucho más efectivos porque no requieren borrar el trazado a medida que se desplazan hacia una posición nueva, y ello aumenta notablemente la velocidad a la cual parecen moverse. En realidad, el desarrollo de los sprites significa que por primera vez es posible escribir juegos recreativos en BASIC, para lo cual antes era imprescindible utilizar el código de lenguaje máquina.

Los principios de los gráficos móviles se pueden utilizar en conjunción con programas sencillos en BASIC. Muchos ordenadores personales poseen órdenes que le permiten al usuario imprimir (PRINT) en posiciones específicas de la pantalla, como PRINT AT en el Spectrum y PRINT @ en el Dragon.

Cómo se crea una estrella

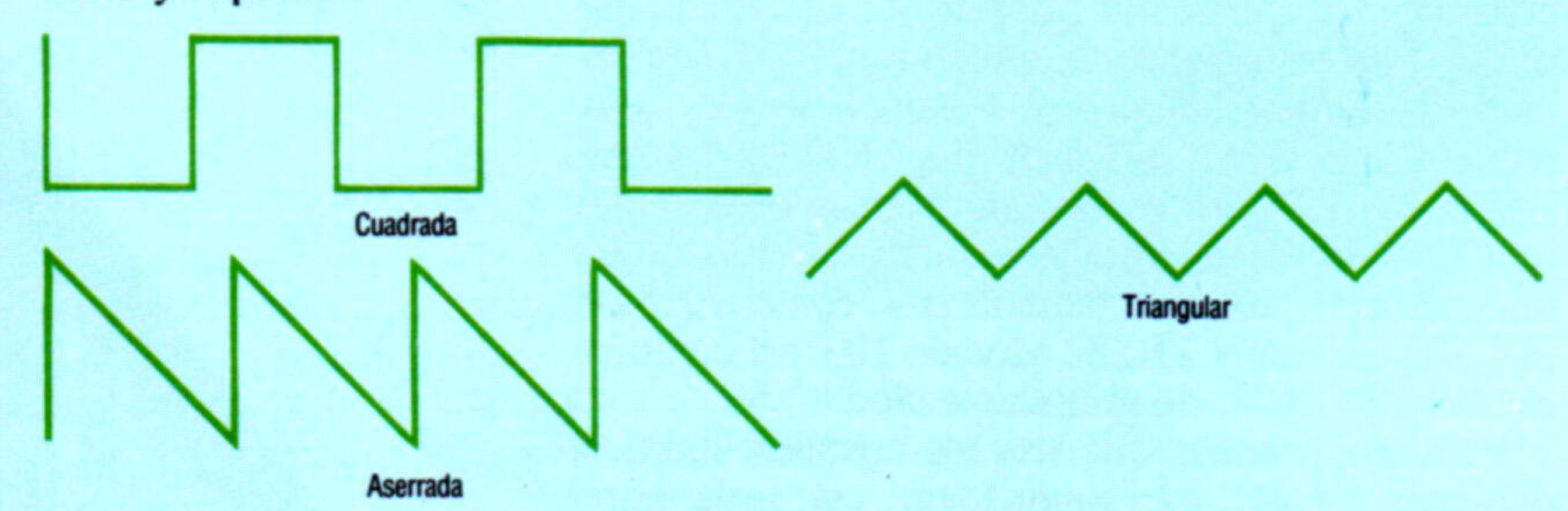
He aquí un breve programa para el Dragon utilizando PRINT. En él, la variable X es la posición en pantalla donde se ha de imprimir la estrella. Observe que la línea 40 borra la estrella antigua mientras se imprime la nueva.

```
10 CLS: REM LIMPIAR PANTALLA
20 FOR X = 160 TO 191
30 PRINT @ X,"*";
40 PRINT @ X – 1," ";
50 NEXT X
60 END
```

Forma de onda

La forma de onda es la "forma" repetitiva de la señal producida por un oscilador (véase p. 247) y le proporciona al sonido su carácter. Dos instrumentos distintos que ejecuten notas a la misma altura no suenan igual, y ello se debe en parte a que las formas de onda son diferentes. Las formas de onda más comunes son la cuadrada (o pulso), la triangular y la aserrada (véase ilustración).

La mayoría de los ordenadores personales proporciona sólo una forma de onda, por lo general del tipo de pulso. Ésta es la razón por la cual muchos de ellos poseen ese sonido sintético inconfundible y áspero.



El Commodore 64 es el ordenador más interesante desde el punto de vista musical, fundamentalmente porque el usuario puede seleccionar cualquiera de las tres formas de onda básicas en tres osciladores distintos. Las formas de onda se pueden modificar utilizando filtros, que alteran el tono de manera muy parecida a la de los controles *bass/treble* de un equipo de alta fidelidad y tienen la cualidad de suavizar el sonido. Aún más útil resulta la capacidad de cambiar estos ajustes de filtro durante la duración de una nota. Esto permite que se puedan imitar sonidos naturales con más exactitud y producir sonidos artificiales más atractivos.

Ruido

El ruido es un tipo de sonido complejo efectuado por vibraciones al azar. El oído no puede captar un patrón repetitivo, por lo cual no oye ninguna altura específica. Imaginemos algunos sonidos cotidianos como la lluvia, el viento y los truenos. Estos ruidos no suenan igual porque son una combinación de ruido puro (impredecible) con algunos tonos dominantes. La mayoría de los ordenadores que poseen capacidad de ruido le permitirán, por tanto, modular el ruido de alguna manera o mezclarlo con notas puras. Los posibles efectos van desde un "viento susurrante" hasta violentas explosiones.

Salida

La salida por lo general se realiza a través del altavoz del televisor. Si éste es el caso, el usuario puede conectar el televisor a su equipo de alta fidelidad mediante una grabadora de video. No obstante, algunos ordenadores sólo pueden dar salida al sonido a través de un pequeño altavoz incorporado. Con estas máquinas es imposible obtener un sonido de buena calidad sin modificar su hardware o comprar un accesorio externo. Puede darse el caso de que el ordenador que usted utiliza posea una salida apta para conectarla directamente a su equipo de alta fidelidad, por lo cual vale la pena el esfuerzo que implica producir formas complejas de sonido.

Cristal líquido

Las visualizaciones en cristal líquido, muy usadas en relojes y calculadoras, comienzan a aparecer en los ordenadores

Las visualizaciones en cristal líquido (LCD: *Liquid Crystal Display*) existen desde finales de 1973, cuando aparecieron por primera vez en las calculadoras. Posteriormente se emplearon en la fabricación de relojes digitales y contribuyeron en gran medida a la popularidad de esa clase de relojes. Ahora las LCD se están haciendo un lugar en la industria de los microordenadores. Se utilizan en máquinas portátiles de tamaño A-4, como el Epson HX-20 y el Tandy TRS80 Modelo 100, así como en el Sharp PC5000, de gran capacidad.

Para comprender qué son los cristales líquidos, debemos señalar, en primer lugar, que toda materia experimenta variaciones en su manera de ser a causa de la temperatura y la presión, que influyen en la mayor o menor cohesión entre sus moléculas. De este modo, pasa, sucesivamente, por tres estados: sólido (o cristalino), líquido y gaseoso. Sólo en el estado sólido se puede hallar una alineación regular de las moléculas de una sustancia. La única excepción la constituyen un pequeño número de sustancias, en las que la alineación regular se mantiene parcialmente en el estado líquido. Estas sustancias, cuya naturaleza es un secreto celosamente guardado, se conocen como *cristales líquidos*.

Hasta mediada la década de los setenta, las visualizaciones de las calculadoras y los relojes estaban compuestas por LED (*Light Emitting Diodes*: diodos emisores de luz) en forma de barra, dispuestos de manera que formaban una versión más bien angular de una letra o un número. Pero las visualizaciones por LED poseen varios inconvenientes: requieren considerables cantidades de energía y su tamaño es relativamente grande.

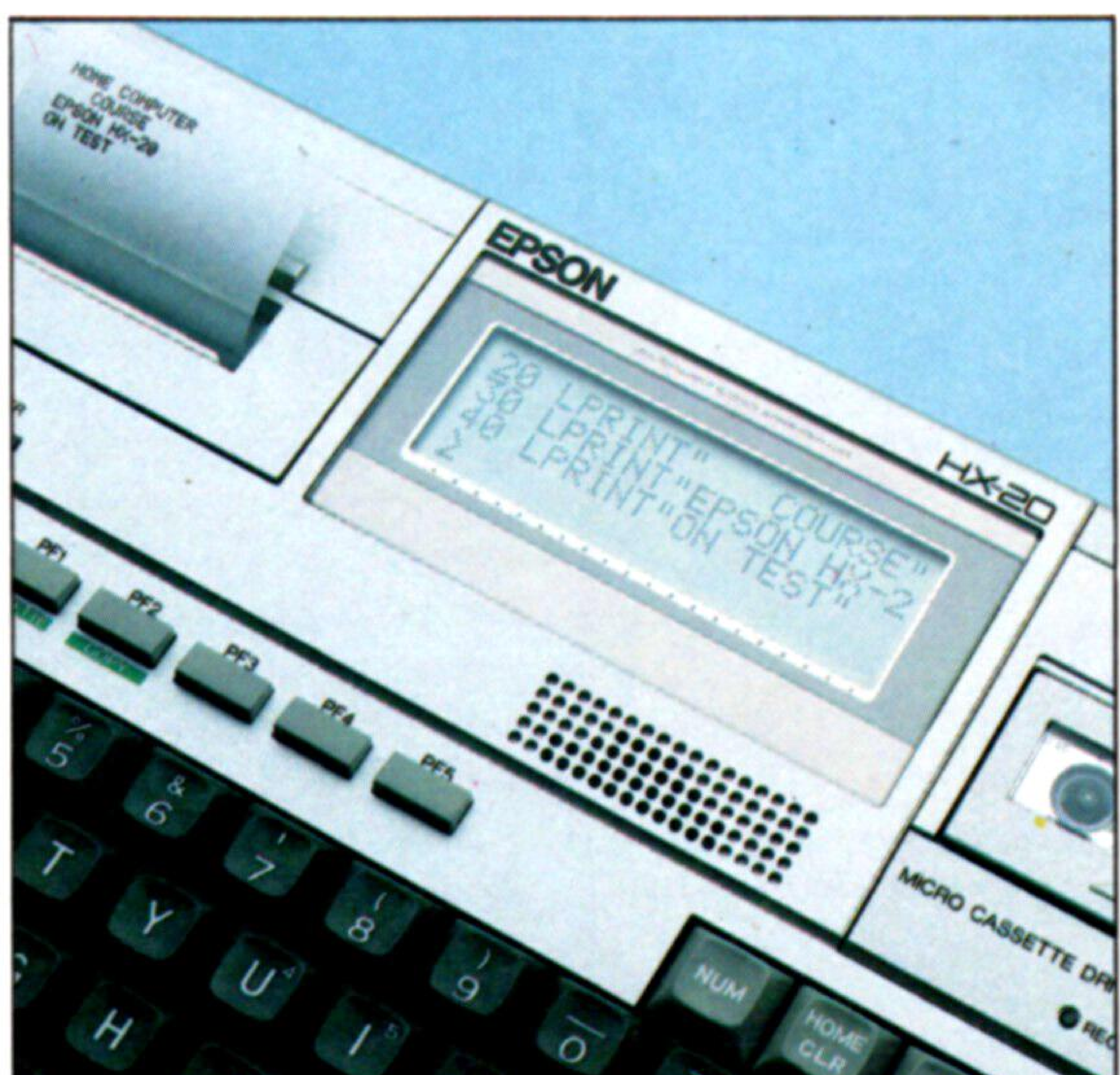
En la búsqueda de métodos alternativos para la visualización de la información, se descubrió que se podía alterar la alineación de las moléculas de los cristales líquidos mediante una corriente eléctrica; y, más aún, que esta alteración era puramente local. Una vez establecido este principio, fue posible construir un soporte para visualizar la información. El primer paso consistió en componer electrodos con la forma de un carácter, en las caras interiores de dos láminas de vidrio. Entre éstas se intercaló una capa muy delgada de cristal líquido y se aplicó un voltaje. A la luz normal pareció que no sucedía nada, pero cuando se aplicaron filtros polarizantes (véase diagrama) en las partes posterior y anterior y se montó toda la estructura contra un fondo reflector, se produjo el efecto deseado: un carácter claramente definido con un fondo neutro.

El proceso en virtud del cual se define este carácter requiere que la luz pase a través del primer filtro y que, de este modo, se polarice verticalmente. Luego se desvía 90°, y a causa de este hecho queda concentrada en el filtro posterior. De esta manera, el área del cristal líquido a la cual se le ha aplicado un voltaje aparece como un área oscura. En la ac-

Filtro de polarización vertical
El filtro de polarización de la parte anterior de la visualización rechaza toda la luz excepto las ondas luminosas que oscilan verticalmente. Incorpora un filtro ultravioleta para prolongar la vida del cristal líquido

Cristal líquido
El cristal líquido propiamente dicho está intercalado entre dos láminas de vidrio selladas por los cantos. Las caras interiores de estas láminas de vidrio están impresas con una "tinta" de óxido de estaño para formar los electrodos

Electrodos negativos
Éste es un "cuadro" de todos los caracteres de la visualización. Todos los componentes están unidos entre sí o "mancomunados"

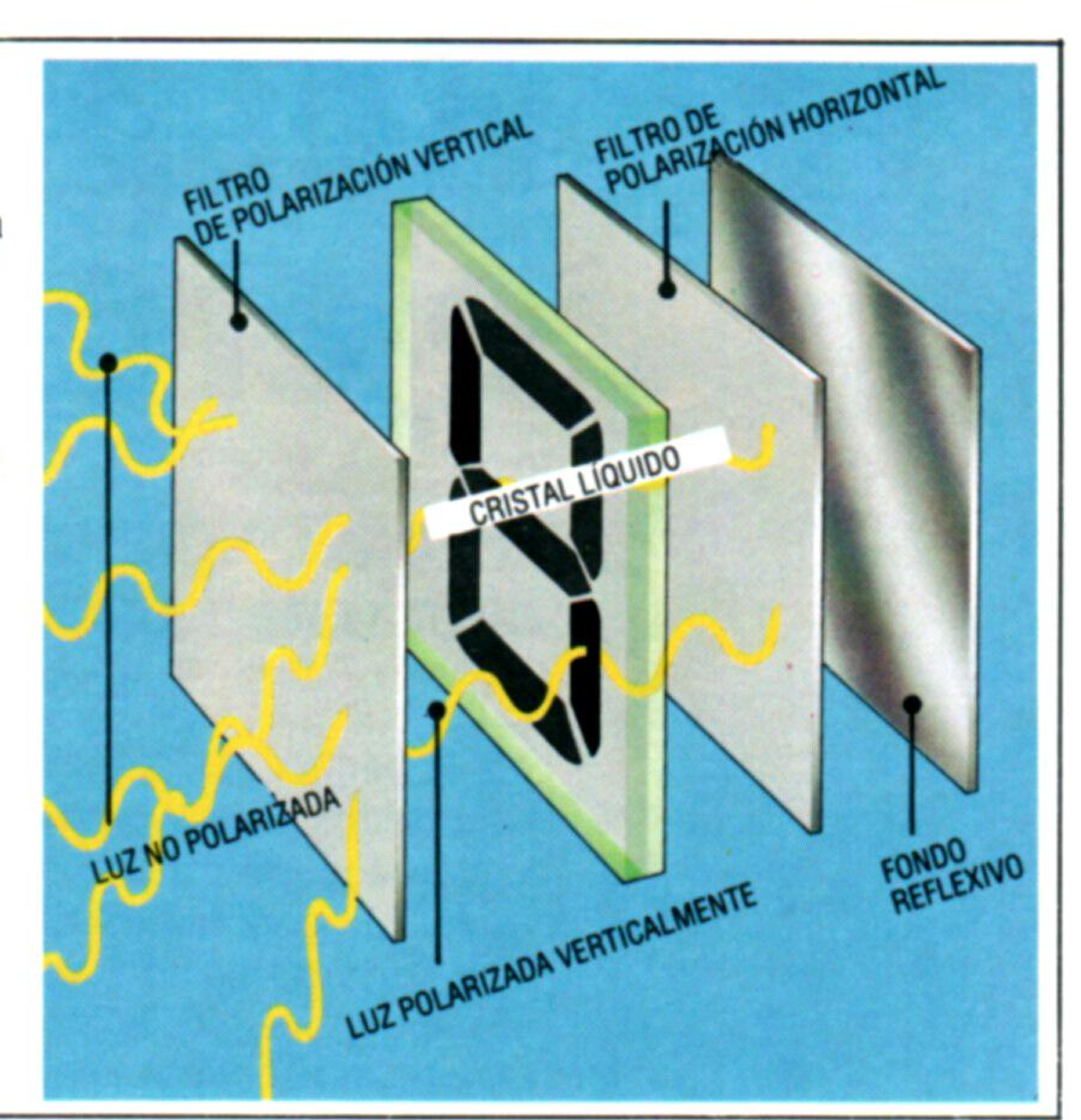


Ángulo de visión
Uno de los refinamientos del Epson HX-20, un ordenador portátil notablemente compacto y sofisticado, es el ajustador del ángulo visual. Los cristales líquidos están compuestos por moléculas largas y delgadas que poseen polos magnéticos situados en el medio de sus lados largos. La aplicación de un voltaje eléctrico a través de su longitud hace que intenten enroscarse por los extremos, mientras su estabilidad natural intenta mantenerlas en su lugar. Cuanta más corriente se les aplique, más se enroscarán

Pasos de conexión
Cada electrodo está conectado a su circuito activador por medio de un depósito casi invisible de tinta conductora, situado en la superficie ya sea del vidrio anterior o del posterior

Filtro de polarización horizontal
El filtro de la parte posterior de la visualización tiene su eje de polarización en 90° respecto al filtro de la parte anterior, de manera que rechaza toda la luz excepto la que oscila sobre el plano horizontal

Polarización
El filtro de la parte anterior de una visualización en cristal líquido sólo permite que pasen a través de él los rayos de luz que tengan la oscilación electromagnética orientada verticalmente. El cristal líquido gira luego la "polarización" en 90°, permitiendo que pase la luz a través del filtro posterior (horizontal) y vuelva a ser reflejada. Cuando se aplica energía a un segmento de la LCD, la polarización deja de rotar, y el resultado es una imagen negra



Chapa reflectora
La mayoría de las LCD se basan en la luz reflejada y, por tanto, están apoyadas sobre una lámina metálica de contraste. Algunas, sin embargo, poseen detrás una fuente de luz

Electrodos positivos
En esta representación de los caracteres, los circuitos activadores pueden direccionar individualmente a cada uno de los componentes, que están separados

tualidad se sigue exactamente el mismo procedimiento para la fabricación de las LCD. Sin embargo, los electrodos están impresos en tinta traslúcida, incolora, sobre la superficie del vidrio, y al secarse quedan casi invisibles.

A causa de que es necesario producir una variedad de caracteres a partir de la misma matriz, aún se utiliza el método original (una combinación de barras cortas que forman caracteres angulares), si bien cada vez están siendo más comunes las LCD matriciales. Debido a que se puede direccionar individualmente cada punto de la matriz, las capacidades para gráficos son muy avanzadas: se pueden producir tanto formas continuas como caracteres para gráficos convencionales.

Teóricamente es posible emplear LCD matriciales direccionadas para que actúen como las pantallas de los receptores de emisiones de televisión, así como monitores para ser utilizados con ordenadores. El principal inconveniente para ello es la carencia de variabilidad del contraste. Éste fue uno de los problemas que se consideraron mientras se desarrollaban los diversos televisores de pantalla plana que están empezando ahora a salir al mercado. Es necesario reducir el tamaño de los elementos individuales de la matriz hasta un nivel aproximado al de un pixel de un tubo de rayos catódicos, con el fin de conseguir una resolución aceptable. Otra posibilidad requeriría la utilización de varias LCD intercaladas entre sí y funcionando de manera simultánea. Este método se emplea habitualmente para la visualización de información compleja que necesita un espacio más amplio.

El tiempo de respuesta de una LCD de gran calidad a la temperatura normal de funcionamiento (20 °C) es de 70 milisegundos aproximadamente para la subida de neutro a negro, y 80 milisegundos más para la bajada a neutralidad otra vez. Este total de 150 milisegundos significa una clara desventaja en comparación con la respuesta de 0,00025 milisegundos del tubo de rayos catódicos. No obstante, la LCD cuenta con ciertas ventajas. Ya hemos analizado su tamaño compacto, pero quizá la mayor ventaja sea su escaso consumo energético. Una LCD típica consume apenas 10 microvatios por centímetro cuadrado de visualización.

Al variar el voltaje que pasa a través de un cristal líquido se produce un efecto muy interesante. La inclinación de las moléculas aumenta y disminuye de acuerdo con el que se aplique. La firma Epson, por ejemplo, utiliza muy bien este efecto en el ordenador portátil HX-20, confiriéndole a la visualización un ajuste del ángulo de visión.

Cuando la luz del sol es intensa se hace evidente una ventaja adicional. El contraste de una visualización por tubo de rayos catódicos disminuye sustancialmente, pero como los cristales líquidos se perciben por la *ausencia* de luz reflejada por ellos, las LCD adquieren mayor contraste y, por lo tanto, son más legibles cuanta más luz exterior haya.

A pesar de que sólo cuenta con diez años de existencia, la tecnología de las visualizaciones en cristal líquido ya ha hecho incursiones significativas en mercados tradicionalmente reservados a los tubos de rayos catódicos. En el futuro, y en la medida en que aumenten las exigencias de microminiaturización, se espera que esta tecnología continúe desarrollándose.

Respuestas a los ejercicios

He aquí algunas soluciones a los problemas de la página 235, si bien usted puede haber encontrado otros métodos alternativos

En la página 235 le propusimos nueve problemas, concebidos para comprobar su habilidad en la utilización de las sentencias y funciones más usadas en BASIC. He aquí las soluciones que sugerimos.

Si ha seguido el curso de programación BASIC desde el principio, quizá haya identificado los ejercicios con problemas con los que ya nos habíamos encontrado antes y que habíamos resuelto. Sus soluciones pueden ser distintas a las nuestras. Es muy raro que exista sólo una manera de resolver un problema; el procedimiento adoptado por usted puede ser tan bueno como el nuestro o incluso mejor.

Si encuentra que las soluciones son tan enigmáticas como las preguntas, vuelva a leer el curso desde la página 149 y vaya estudiando las soluciones a medida que avance. Si ha logrado resolver los ejercicios 2, 4, 6 y 8, ha comprendido la mayoría de las lecciones de nuestro curso.

```
100 REM EJERCICIO DE REVISION 1
200 INPUT "DIGITE CUALQUIER NUMERO";A
300 INPUT "DIGITE OTRO NUMERO";B
400 LET C = A + B
500 PRINT "SU SUMA ES";C
```

```
100 REM EJERCICIO DE REVISION 2
200 LET A$ = "PRIMERA PALABRA"
300 LET B$ = "SEGUNDA PALABRA"
400 LET C$ = A$ + B$
500 PRINT C$
```

```
100 REM EJERCICIO DE REVISION 3
200 INPUT "DIGITE CUALQUIER PALABRA";P$
300 LET L = LEN(P$)
400 PRINT "LA PALABRA QUE HA DIGITADO
    TIENE";L;"CARACTERES"
```

```
100 REM EJERCICIO DE REVISION 4
200 PRINT "PULSE CUALQUIER TECLA"
300 FOR C = 0 TO 1 STEP 0
400 LET A$ = INKEY$
500 IF A$ <> " " THEN LET C = 2
600 NEXT C
700 PRINT "EL VALOR ASCII DE";A$;
    "ES";ASC(A$)
```

Véase "Complementos al BASIC", de pp. 175 y 215. En el Spectrum, reemplazar la línea 700 por:

```
700 PRINT "EL VALOR ASCII DE";A$;
    "ES";CODE(A$)
```

```
100 REM EJERCICIO DE REVISION 5
200 INPUT "DIGITE UNA PALABRA";P$
300 LET L$ = RIGHT$(P$,1)
400 PRINT "LA ULTIMA LETRA DE LA
    PALABRA ERA ";L$
```

Véase el recuadro "Complementos al BASIC" de p. 149. En el Spectrum este ejercicio se lee:

```
100 REM EJERCICIO DE REVISION 5
200 INPUT "DIGITE UNA PALABRA";P$
250 LET N = LEN(P$)
300 LET L$ = P$(N)
400 PRINT "LA ULTIMA LETRA DE LA
    PALABRA ERA";L$
```

```
100 REM EJERCICIO DE REVISION 6
200 PRINT "DIGITE UN NOMBRE EN LA FORMA:"
300 PRINT "NOMBRE Y PRIMER APELLIDO"
400 PRINT "P. EJ. ROSA TORRES"
500 INPUT "NOMBRE";N$
600 LET S = 0:LET L = LEN(N$)
700 FOR P = 1 TO L
800 IF MID$(N$,P,1) = " " THEN LET S = P
900 NEXT P
950 PRINT "EL ESPACIO ERA EL";S;"º CARACTER"
```

Véase "Complementos al BASIC", p. 149. En el Spectrum, reemplazar la línea 800 anterior por:

```
800 IF N$(P) = " " THEN LET S = P
```

```
100 REM EJERCICIO DE REVISION 7
150 LET X$ = "º"
200 PRINT "DIGITE UN NOMBRE EN LA FORMA:"
300 PRINT "NOMBRE Y PRIMER APELLIDO"
400 PRINT "P. EJ. ROSA TORRES"
500 INPUT "NOMBRE";N$
600 LET S = 0: LET L = LEN(N$)
700 FOR P = 1 TO L
800 IF MID$(N$,P,1) = " " THEN LET S = P
900 NEXT P
925 IF S = 3 THEN LET X$ = "ER"
950 PRINT "EL ESPACIO ERA EL";S;X$;
    "CARACTER"
```

```
100 REM EJERCICIO DE REVISION 8
200 INPUT "DIGITE UNA ORACION";O$
300 LET C = 1
400 FOR P = 1 TO LEN (O$)
500 IF MID$(O$,P,1) = " " THEN LET C = C + 1
600 NEXT P
700 PRINT "LA ORACION QUE DIGITO
    TENIA";C;"PALABRAS"
```

Véase "Complementos al BASIC", p. 149. En el Spectrum, reemplazar la línea 500 anterior por:

```
500 IF O$(P) = " " THEN LET C = C + 1
```

```
100 REM EJERCICIO DE REVISION 9
200 FOR C = 128 TO 255
300 PRINT "CARACTER N.º";C;"=";CHR$(C)
400 REM BREVE DEMORA AQUI
500 FOR D = 1 TO 500
600 NEXT D
700 REM FIN DE LA DEMORA
800 NEXT C
```

DONDE CONSEGUIR TU sinclair

ALAVA

COMPONENTES ELECTRONICOS GAZTEIZ
Domingo Beltran, 58 (Vitoria)
DEL CAZ
Avda. Gazteiz, 58 (Vitoria)
VALBUENA
Virgen Blanca, 1 (Vitoria)

ALBACETE

ELECTRO MIGUEL
Tesifonte Gallego, 27
TECON
Maria Marin, 13

ALICANTE

ASEMCA (Villena)
Avda. de la Constitucion, 54 (Villena)
CONSULTING DESARROLLO INFORMATICO
Pais Valencia, 54 (Alcoy)
COMPONENTES ELECTRONICOS LASER
Jaime M.ª Buch, 7
ELECTRODATA LEVANTE
San Vicente, 28
ELECTRONICA AITANA
Limones, s/n. Edificio Urgull (Benidorm)
ELECTRONICA OHMIO
Avda. El Hamed, 1
LIBRERIA LLORENS
Alameda, 50 (Alcoy)

AVILA

FELIX ALONSO
San Segundo, 15

BADAJOZ

MECANIZACION EXTREMEÑA
Vicente Barantes, 18
SONYTEL
Villanueva, 16

BARCELONA

ARTO
C/ Angli, 43
BERENGUERAS
C/ Diputacion, 219
CATALANA D'ORDINADORS
C/ Trafalgar, 70
CECSA
C/ Mallorca, 367
COMPUTERLAND
C/ Infanta Carlota, 89
COMPUTERLAND
Trav. de Dalt, 4
COPIADUX
C/ Dos de Mayo, 234
D. P. 2000
C/ Sabino de Arana, 22-24
DIOTRONIC
C/ Conde Borrell, 108
EL CORTE INGLES
Avda. Diagonal, 617-619
EL CORTE INGLES
Pza. Cataluña, 14
ELECTRONICA H. S.
C/ S. José Oriol, 9
ELECTRONICA SAUQUET
C/ Guilleries, 10
ELEKTROCOMPUTER
Via Augusta, 120
EXPOCOM
C/ Villarroel, 68
GUIBERNAU
C/ Sepulveda, 104
INSTA-DATA
P.º S. Juan, 115
MAGIAL
C/ Sicilia, 253
MANUEL SANCHEZ
Pza. Major, 40 (Vic)
MILLIWATTS
C/ Melendez, 55 (Mataro)
ONDA RADIO
Gran Via, 581
RADIO ARGANY
C/ Borrell, 45
RADIO SONDA
Avda. Abad Margat, 77 (Tarrasa)
RAMEL ELECTRONICA
Cr. de Vic, 3 (Manresa)
REDISA GESTION
Avda. Sarria, 52-54
RIFE ELECTRONICA
C/ Aribau, 80, 5.º, 1.ª
SERVICIOS ELECTRONICOS VALLES
Pza. del Gas, 7 (Sabadell)
SISTEMA
C/ Balmes, 434
S. E. SOLE
C/ Muntaner, 10
SUMINISTROS VALLPARADIS
C/ Dr. Ferrén, 172 (Tarrasa)
TECNOHIFI, S. A.
C/ La Rambleta, 19
VIDEOCOMPUT
P.º Pep Ventura, 9, Bl. C. Bjos. Bis (Vic)

BURGOS

COMELECTRIC
Calzada, 7
ELECTROSON
Conde don Sancho, 6

CACERES

ECO CACERES
Diego Maria Crehuet, 10-12

CADIZ

ALMACENES MARISOL
Camoens, 11 (Ceuta)
INFORSA
Avda. Fuerzas Armadas, 1 (Algeciras)
ELECTRONICA VALMAR
Ciudad de Santander, 8
M. R. CONSULTORES
Multi Centro Merca 80 (Jerez de la Frontera)
PEDRO VAREA
Porvera, 36 (Jerez de la Frontera)
LEO COMPUTER
Garcia Escamez, 3
SONYTEL
Queipo de LLano, 17
SONYTEL
Jose Luis Diez, 7
T L C Y AUTOMATICA
Dr. Herrera Quevedo, 2

CASTELLON

NOU DESPACH'S
Rey D. Jaime, 74

CIUDAD REAL

COMERCIAL R. P.
Travesera de Coso, 2 (Valdepeñas)
ECO CIUDAD REAL
Calatrava, 8

CORDOBA

ANDALUZA DE ELECTRONICA
Felipe II, 15
CONTROL
Conde de Torres Cabrera, 9
ELECTRONICA PADILLA
Sevilla, 9
MORM
Plaza Colon, 13
SONYTEL
Arte, 3
Avda. de los Mozarabes, 7

CUENCA

SONYTEL
Dalmacio Garcia Izcara, 4

GERONA

AUDIFILM
C/ Albareda, 15
CENTRE DE CALCUL DE CATALUNYA
C/ Barcelona, 35
S. E. SOLE
C/ Sta. Eugenia, 59

GRANADA

INFORMATICA Y ELECTRONICA
Melchor Almagro, 8
SONYTEL
Manuel de Falla, 3
TECNIGAR
Ancha de Gracia, 11

GRANOLLERS

COMERCIAL CLAPERA
C/ Maria Maspons, 4

GUIPUZCOA

ANGEL IGLESIAS
Sancho el Sabio, 7-9
BHP NORTE
Ramon M.ª Lili, 9
ELECTROBON
Reina Regente, 4

HUELVA

SONYTEL
Ruiz de Alda, 3

HUESCA

ELECTRONICA BARREU
M.ª Auxiliadora, 1

IBIZA

IBITEC
C/ Aragon, 76

JAEN

CARMELO MILLA
Coca de la Piñera, 3
MARA ILUMINACION
Avda. Linares, 13 (Ubeda)
MICROJISA
Garcia Rebull, 8
SONYTEL
Jose Luis Diez, 7
SONYTEL
Pasaje del Generalisimo, 3 (Linares)

LA CORUÑA

DAVIÑA
Republica de El Salvador, 29 (Santiago)
PHOTOCOPY
Teresa Herrera, 9
SONYTEL
Avda. de Arteijo, 4
SONYTEL
Tierra, 37

LAS PALMAS DE GRAN CANARIA

COMPUTERLAND
Carvajal, 4
CHANRAI
Triana, 3
EL CORTE INGLES
Jose Mesa y Lopez, 18

LEON

ELECTROSON
Avda. de la Facultad, 15
MICRO BIERZO
Carlos I, 2 (Ponferrada)
RADIO RACE
Modesto Lafuente, 3

LERIDA

SELEC
C/ Ferrer y Busquet, 14 (Mollerusa)
SEMIC
C/ Pi y Margall, 47

LUGO

ELECTROSON
Concepcion Arenal, 38
SONYTEL
Primo de Rivera, 30

MADRID

ALFAMICRO
Augusto Figueroa, 16
BELLTON'S
Torpedero Tucuman, 8
CHIPS-TIPS
Pto. Rico, 21
CMP
Pto. Santa Maria, 128
COMPUTERLAND
Castello, 89
COSESA
Barquillo, 25
DINSA
Gaztambide, 4
DISTRIBUIDORA MADRILEÑA
Todos sus centros
ELECTROSON
Duque de Sexto, 15 (y otros centros)
INVESMICROSTORE
Genova, 7
J.P. MICROCOMPUT
Montesa, 44
EL CORTE INGLES
Todos sus centros
ELECTRONICA SANDOVAL
Sandoval, 4
PENTA
Dr. Cortezo, 12
RADIO CINEMA
Antonio Acuña, 3
RADIO QUER
Todos sus centros
SONYTEL
Clara del Rey, 24 (y todos sus centros)
SONICAR
Vallehermoso, 19
VIDEOMUSICA
Orense, 28

MALAGA

EL CORTE INGLES
Prolongación Alameda, s/n.
INGESCON
Edificio Galaxia
SONYTEL
Salitre, 13

MELILLA

OFI-TRONIC
Hermanos Cayuela, 11

MENORCA

ELECTRONICA MENORCA
C/ Miguel de Veri, 50 (Mahon)

MURCIA

COMPUTER LIFE
Alameda San Anton, 2 (Cartagena)
EL CORTE INGLES
Libertad, 1
ELECTRONICA COMERCIAL CRUZ
Rio Segura, 2
MICROIN
Gran Via, 8

NAVARRA

ENER
Paulino Caballero, 39
GABINETE TECNICO EMPRESARIAL
Juan de Labrit, 3
JOSE LUIS DE MIGUEL
Arrieta, 11 bis

OVIEDO

AUTECA
Valentin Masip, 25
EDIMAR
Cangas de Onis, 4-6 (Gijon)
ELECTRONICA RATO
Versalles, 45 (Aviles)
RADIO NORTE
Uria, 20
RESAM ELECTRONICA
San Agustin, 12 (Gijon)
RETELCO
Cabrales, 31 (Gijon)
SELECTRONIC
Fermin Canellas, 3

ORENSE

SONYTEL
Concejo, 11

PONTEVEDRA

EL CORTE INGLES
Gran Via, 25 (Vigo)
ELECTROSON
Santa Clara, 32
ELECTROSON
Venezuela, 32 (Vigo)
SONYTEL
Salvador Moreno, 27
SONYTEL
Gran Via, 52 (Vigo)
TEFASA COMERCIAL
San Salvador, 4 (Vigo)

PALMA DE MALLORCA

GILFT
Via Alemania, s/n
IAM
C/ Cecilio Metlo, 5
TRON INFORMATICA
C/ Juan Alcover, 54, 6.º C

LA RIOJA

YUS COMESSA
Cigueña, 15

SALAMANCA

DEL AMO
Arco, 5
PRODISTELE
España, 65

SANTANDER

LAINZ S. A.
Reina Victoria, 127
RADIO MARTINEZ
Dr. Jimenez Diaz, 13

SEGOVIA

ELECTRONICA TORIBIO
Obispo Quesada, 8

SEVILLA

A.D.P.
San Vicente, 3
EL CORTE INGLES
Duque de la Victoria, 10
SCI
Aceituno, 8
SONYTEL
Pages del Corro, 173
Adriano, 32

TARRAGONA

AIA
Ramblá Nova, 45, 1.º
CIAL. INFORMATICA TARRAGONA
C/ Gasometro, 20
ELECTRONICA REUS
Avda. Prat de la Riba, 5 (Reus)
SEIA
Rambla Vella, 7 B
S.E. SOLE
C/ Cronista Sese, 3
T. V. HUGUET
Pza. Major, 14 (Montblac)
VIRGILI
C/ Dr. Gimbernat, 19 (Reus)

STA. CRUZ DE TENERIFE

COMPUTERLAND
Mendez Nuñez, 104 B
TRENT CANARIAS
Serrano, 41

VALENCIA

ADISA
San Vicente, 33 (Gandia)
CESPEDES
San Jacinto, 6
COMPUTERLAND
Marques del Turia, 53
DIRAC
Blasco Ibañez, 116
EL CORTE INGLES
Pintor Sorolla, 26
Melendez Pidal, 15
PROMOCION INFORMATICA
Pintor Zariñena, 12

VALLADOLID

SONYTEL
Leon, 4

VIZCAYA

BILBOMICRO
Aureliano del Valle, 7
DATA SISTEMAS
Henao, 58
DISTRIBUIDORA COM
Gran Via, 19-21 y todos sus centros
EL CORTE INGLES
Gran Via, 9
ELECTROSON
Alameda de Urquijo, 71
San Vicente, 18 (Baracaldo)
GESCO INFORMATICA
Alameda de Recalde, 76
KEYTRON
Hurtado de Amezaga, 20

ZAMORA

MEZZASA
Victor Gallego, 17

ZARAGOZA

EL CORTE INGLES
Sagasta, 3
SONYTEL
Via Pignatelli, 29-31



DISTRIBUIDOR EXCLUSIVO:
INVESTRONICA

Central Comercial
TOMAS BRETON, 60
TELF. 468 03 00
TELEX 23399 IYCO E
MADRID

Delegación Cataluña
MUNTANER, 565
TELF. 212 68 00
BARCELONA